

Delta-Sigma Signalverarbeitung in der Regelungstechnik

Von der Fakultät für Elektrotechnik, Informationstechnik, Physik
der Technischen Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung des Grades eines Doktors
der Ingenieurwissenschaften (Dr.-Ing.)

genehmigte Dissertation

von M. Sc. Axel Klein

aus Filderstadt

eingereicht am: 28.03.2019

mündliche Prüfung am: 10.09.2019

1. Referent: Prof. Dr.-Ing. Walter Schumacher

2. Referent: Prof. Dr. rer. nat. Meinhard Schilling

Druckjahr: 2019

Vorwort

Die vorliegende Dissertation entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Regelungstechnik der Technischen Universität Braunschweig.

Mein außerordentlicher Dank gilt meinem Doktorvater Herrn Prof. Dr.-Ing. Walter Schumacher für die Förderung meiner wissenschaftlichen Arbeit, die fachlichen Gespräche und die Freiräume bei der Gestaltung dieser Arbeit. Herrn Prof. Dr. rer. nat. Meinhard Schilling danke ich für die Übernahme der Mitberichterstattung, das Interesse an meiner Arbeit sowie die wertvollen fachlichen Hinweise zu dieser Dissertation. Herrn Prof. Dr.-Ing. Markus Henke danke ich für die Übernahme des Prüfungsvorsitzes.

Mein herzlicher Dank gilt meinen ehemaligen Kollegen am Institut für Regelungstechnik, die durch ihre Hilfsbereitschaft und durch freundschaftlichen Umgang in hohem Maße zum Gelingen dieser Arbeit beigetragen haben. Besonders möchte ich Dr.-Ing. Michael Homann für die wertvollen Anregungen zur initialen Ausrichtung meiner wissenschaftlichen Arbeit und die fortwährende fachliche Diskussion und Zusammenarbeit danken. Weiter möchte ich Herrn Malte Thielmann für die intensive fachliche und praktische Zusammenarbeit bedanken.

Weiterhin möchte ich mich bei den Hilfswissenschaftlern und den Studenten bedanken, die bei ihren Abschlussarbeiten von mir betreut wurden. Sie bereicherten meine Arbeit mit neuen Ansichten und erarbeiteten notwendige Lösungen. Besonders hervorheben möchte ich die längerfristig am Institut tätig gewesenen Studenten Dean Lancaster, Max Brüggemann, Lasse Cordes und Richard Mathis.

Zum Schluss möchte ich meinen größten Dank meiner Familie aussprechen, die mich immer während dieser Arbeit unterstützt und motiviert hat.

Inhaltsverzeichnis

Abbildungsverzeichnis	V
Tabellenverzeichnis	IX
Kurzfassung	XIII
Abstract	XIV
1 Einleitung	1
1.1 Stand der Forschung	1
1.2 Zielsetzung der Arbeit	2
2 Delta-Sigma Modulator	4
3 Delta-Sigma Signal Verarbeitung	8
3.1 Lineare Bitstrom Operationen	8
3.1.1 Logik Operation	9
3.1.2 Delta Addierer	10
3.1.3 Bitstrommodifikation	11
3.1.4 Zählerbasiert	14
3.1.5 Quantendekodierung	16
3.1.6 Erweiterte Quantendekodierung	17
3.1.7 Bipolare Interpretation	19
3.1.8 Evaluation	20
3.1.9 Ergebnisse	21
3.2 Nichtlineare Bitstrom Operationen	25
3.2.1 Schnelle Filter	26
3.2.2 Multiplikation	27
3.2.2.1 Delta Addierer	28
3.2.2.2 Delta Addierer Modifiziert	29
3.2.2.3 Logische Operationen	30
3.2.2.3.1 Statistische Betrachtung	30
3.2.2.3.2 Mathematische Funktionen	40
3.2.2.3.3 Standardabweichung	43
3.2.2.3.4 Quantisierungsrauschen	46
3.2.2.3.5 Ergebnisse	48
3.2.2.4 Evaluation	51

3.2.2.5	Ergebnisse	53
3.2.3	Begrenzung	55
3.2.3.1	Schieberegister	55
3.2.3.2	Ergebnisse	56
3.3	Kaskadierte Operationen	58
4	Delta-Sigma PWM Modulator	61
4.1	Prüfstand	63
4.2	Operationsmodi	64
4.3	Schaltfrequenz	69
4.4	Schaltfrequenzregelung	72
5	Anwendungen	79
5.1	Stromregelung	79
5.1.1	Proportional Stromregler im dreiphasigen System . .	80
5.1.2	Proportional Integral Stromregler im 120° System .	82
5.1.3	Proportional Integral Stromregler im d/q System . .	84
5.1.4	Versuch	86
5.1.5	Ergebnisse	87
5.2	Harmonische Regelung	90
5.2.1	Resonante Regelungsstrukturen	90
5.2.2	Evaluation	92
5.2.3	Ergebnisse	92
5.3	Drehgeberlose Geschwindigkeitsregelung	96
5.3.1	Schätzung der elektromotorischen Kraft	96
5.3.2	Regelkreis	99
5.3.3	Ergebnisse	99
6	Zusammenfassung und Ausblick	103
	Literatur	106
	Veröffentlichungen im Rahmen dieser Dissertation	109
A	Anhang	111
A.1	Parameter der Delta-Sigma Modulatoren	111
A.2	Lösung der PWM-Schaltfrequenz	112

Abbildungsverzeichnis

2.1	$\Delta\Sigma$ -Modulator ($\Delta\Sigma$ -M) erster Ordnung mit Tiefpassfilter [KTS18a]	4
2.2	Betrag der Signalübertragungsfunktion (SÜF) und der Rauschübertragungsfunktion (RÜF) eines $\Delta\Sigma$ -M erster Ordnung und eines $\Delta\Sigma$ -M zweiter Ordnung [HKKS16].	5
2.3	Allgemeine Darstellung eines $\Delta\Sigma$ -M zweiter Ordnung nach [34].	6
3.1	Grundsätzlicher Aufbau einer $\Delta\Sigma$ Signalverarbeitung ($\Delta\Sigma$ SV) Operation mit den drei Verarbeitungsschritten und der äquivalenten DSV Operation [KS18].	9
3.2	Delta Addierer aus Logikbausteinen [17] und das vereinfachte Blockschaltbild [KS18].	10
3.3	Äquivalentes regelungstechnisches Blockschaltbild eines Delta Addierers [HKKS16].	11
3.4	Skalierung basierend auf Delta Addierern, wobei α_i dem i. Bit des Skalierungsfaktors vom höchstwertigen Bit entspricht [17].	11
3.5	Addition zweier Bitströme mit Bitstrommodifikation und die detaillierte Darstellung von -0,25 [KS18].	12
3.6	Addition eines festen Offsets zu einem Bitstrom [KS18].	13
3.7	Zählerbasierte Addition erster Ordnung [KS18].	14
3.8	Zählerbasierte Addition zweiter Ordnung [KS18].	15
3.9	Nullbitstrom Generator basierend auf einem $\Delta\Sigma$ -M erster Ordnung und die vereinfachte Implementierung [KS18].	16
3.10	Die $\Delta\Sigma$ SV Operation Addition in Quantendekodierung [KS18].	17
3.11	Die $\Delta\Sigma$ SV Operation Addition implementiert mit erweiterter Quantendekodierung [KS18].	19
3.12	Basisimplementierung einer $\Delta\Sigma$ SV Operation in Bipolarer Interpretation [KS18].	20
3.13	Die $\Delta\Sigma$ SV Operation Addition implementiert in Bipolarer Interpretation [KS18].	20
3.14	Signallaufplan für die Evaluation von $\Delta\Sigma$ SV Operationen [KS18].	21

3.15	Signalqualität in SINAD nach unterschiedlichen $\Delta\Sigma$ SV Implementierungen einer Sinusschwingung mit variabler Amplitude [KS18].	22
3.16	Allgemeine Darstellung der Integration in $\Delta\Sigma$ SV [KS18]. . .	25
3.17	Allgemeine Implementierung einer nichtlinearen $\Delta\Sigma$ SV Funktion mit schnellem Filter [KS18].	26
3.18	Multiplikation zweier Bitströme mit schnellen Filtern und DSV Operation [KS18].	27
3.19	Limitierung eines Bitstroms mit schnellem Filter und DSV Operation [KS18].	27
3.20	Ausgangsspektrum von $\Delta\Sigma$ -M erster und zweiter Ordnung und der im Zeitbereich quadrierten Signale [KS18].	28
3.21	Multiplikation zweier Bitströme basierend auf Delta Addieren [20].	29
3.22	Modifizierte Multiplikation zweier Bitströme basierend auf Delta Addieren [KS18].	30
3.23	Erwartete Wahrscheinlichkeit einer Eins im Ausgangsbitstrom für die Eingangsbitströme BS_1 und BS_2 , mit der Wahrscheinlichkeit von Einsen $p_{1 2}$, mit OR verknüpft [Kle18].	34
3.24	Erwartete Wahrscheinlichkeit einer Eins im Ausgangsbitstrom für die Eingangsbitströme BS_1 und BS_2 , mit der Wahrscheinlichkeit von $p_{1 2}$ für eine Eins, die mit AND verknüpft wurden [Kle18].	36
3.25	Erwartete Wahrscheinlichkeit einer Eins im Ausgangsbitstrom für die XOR verknüpften Eingangsbitströme BS_1 und BS_2 , mit der Wahrscheinlichkeit von $p_{1 2}$ für eine Eins [Kle18].	39
3.26	Abbild der Funktion $BS_1 + BS_2 - BS_1 \cdot BS_2$ [Kle18].	41
3.27	Abbild der Funktion $BS_1 \cdot BS_2$ [Kle18].	42
3.28	Abbild der Funktion $-BS_1 \cdot BS_2$ [Kle18].	43
3.29	Standardabweichung für die Wahrscheinlichkeit einer Eins im BS_{aus} für die BS_1 und BS_2 , mit der Wahrscheinlichkeit von Einsen $p_{1 2}$, verknüpft mit OR oder AND [Kle18]. . . .	44
3.30	Standardabweichung für die Wahrscheinlichkeit einer Eins im BS_{aus} für die BS_1 und BS_2 , mit der Wahrscheinlichkeit von Einsen $p_{1 2}$, verknüpft mit XOR [Kle18].	45
3.31	Lineares Modell des Quantisierers eines $\Delta\Sigma$ -M, bestehend aus einer Verstärkung (k_q) und einer weißen Rauschquelle ($e(n)$) [28].	46
3.32	Simulation zur Untersuchung der Verkopplung des Rauschens von $\Delta\Sigma$ -M [Kle18].	47
3.33	Versuchsaufbau zur Bewertung der aus logischen Operationen resultierenden Funktionen [Kle18].	49

3.34	Abbildungen durch die Logikoperationen OR und AND und die entsprechenden Operationen $f_{or} = BS_1 + BS_2 - BS_1 \cdot BS_2$ und $f_{and} = BS_1 \cdot BS_2$ [Kle18].	50
3.35	Abbildungen der Logikoperation XOR und die entsprechende mathematische Funktion $f_{xor} = -BS_1 \cdot BS_2$ [Kle18].	51
3.36	Testaufbau für nichtlineare Funktionen mit optionalem zweiten Eingang.	52
3.37	Tiefpass gefilterte Ergebnisse der Operation Multiplikation mit einer Dezimierungsrate von 200 [KS18].	54
3.38	Positive Begrenzung eines Bitstroms basierend auf Schieberegistern [KS18].	56
3.39	Negative Begrenzung eines Bitstroms basierend auf Schieberegistern [KS18].	56
3.40	Tiefpass gefilterte Ergebnisse der Begrenzung mit einer Dezimierungsrate von 20 [KS18].	57
3.41	Versuch zur Evaluierung von kaskadierten $\Delta\Sigma$ SV Operationen.	58
3.42	Signalqualität von kaskadierten $\Delta\Sigma$ SV Operationen [HKS16].	59
4.1	$\Delta\Sigma$ -PWM mit interner Spannungsregelung [10].	61
4.2	Amplituden- und Phasenhysterese des $\Delta\Sigma$ -PWM [KTS18b].	63
4.3	Theoretische Regelfehlertrajektorie im Ausschnitt des Umrichtersechsecks [KTHS17].	66
4.4	Online Indikatoren für Betriebsart- oder Lastpunkterkennung über m aus der Simulation. $\varphi_{soll} = 30^\circ$, $ \underline{e} > N_{2,c}$ mindestens einmal pro SVM-Zyklus [KTHS17].	68
4.5	Erwartete mittlere Geschwindigkeit des Regelfehlers bei unterschiedlichen Schaltzuständen [KTHS17].	69
4.6	Natürliche Schaltfrequenz des $\Delta\Sigma$ -PWM über der Aussteuerung [KTS18b].	72
4.7	Schaltfrequenzregelung des $\Delta\Sigma$ -PWM mit Filter [KTS18b].	74
4.8	Schaltfrequenzen des $\Delta\Sigma$ -PWM bei unterschiedlichen Aussteuerungen [KTHS17].	75
4.9	Geregelte Schaltfrequenz (16 kHz) des $\Delta\Sigma$ -PWM über der Aussteuerung [KTS18b].	75
4.10	Blockschaltbild des minimalen Schaltfrequenzwächters [KTS18b].	76
4.11	Schaltfrequenz mit Schaltfrequenzwächter (10 kHz) des $\Delta\Sigma$ -PWM über der Aussteuerung [KTS18b].	77
4.12	Geregelte Schaltfrequenz (16 kHz) mit Schaltfrequenzwächter (10 kHz) des $\Delta\Sigma$ -PWM über der Aussteuerung [KTS18b].	78
5.1	$\Delta\Sigma$ -PWM mit interner Spannungsregelung und P Stromregelung [10].	81

5.2	$\Delta\Sigma$ -PWM mit interner Spannungsregelung und PI Stromregelung im 120° System.	83
5.3	$\Delta\Sigma$ -PWM mit interner Spannungsregelung und PI Stromregelung im d/q System.	85
5.4	Bodediagramme der Stromregelungen.	88
5.5	Signalqualität der Stromregelung über dem Aussteuerungsgrad.	89
5.6	Bodediagramm eines PI-Reglers und eines resonanten Reglers [MTK18].	91
5.7	Resonanter Regler mit proportionaler Regler und überschwingungs Reglern [15].	92
5.8	Regelkreis mit Führungsgrößengenerator (FG) zur Evaluation der resonanten Regler (PMR) in $\Delta\Sigma$ SV [MTK18]. . . .	93
5.9	Regelkreis mit Führungsgrößengenerator (FG) zur Evaluation der resonanten Regler (PMR) in DSV [MTK18]. . . .	94
5.10	Oberschwingungsbehaftete Stromregelung mit DSV [MTK18].	94
5.11	Oberschwingungsbehaftete Stromregelung mit $\Delta\Sigma$ SV [MTK18].	95
5.12	Gemessene Sollspannungen bei proportionaler Phasenstromregelung mit EMK Vorsteuerung [KTS18a].	97
5.13	Gemessener relativer Stromregelfehler bei proportionaler Phasenstromregelung mit EMK Vorsteuerung [KTS18a]. . .	97
5.14	Drehwinkelschätzung für die $\Delta\Sigma$ -PWM basierend auf dem Regelfehler der proportionalen Stromregler mittels PLL [KTS18a].	98
5.15	$\Delta\Sigma$ -PWM Geschwindigkeitsregelung basierend auf beobachtetem Winkel und Drehzahl einer PLL [KTS18a].	100
5.16	Gemessener Winkel (Hall-Sensor) und mittels PLL beobachteter Winkel [KTS18a].	101
5.17	Drehgeberlose Geschwindigkeitsregelung (gemittelt über 10 Datenpunkte) [KTS18a].	102

Tabellenverzeichnis

3.1	NOT	9
3.2	Wahrheitstabelle des Quantisierers für zählerbasierte Operationen [KS18].	15
3.3	Wahrheitstabelle eines Bitstroms (BS) minus den NBS ergibt das quantendekodierte Signal (QD) [29].	16
3.4	Wahrheitstabelle zweier Bitströme (BS_x) und deren “Nullbitströme” (ZBS_x), die quantendekodierten Werte (QD_x), sowie die Ergebnisse von Subtraktion und Addition der quantendekodierten Werte als auch der direkt verarbeiteten Bitströme. In den hervorgehobenen Zeilen sind die “Nullbitströme” identisch [KS18].	18
3.5	Gruppenlaufzeit der Operationsklassen in Taktzyklen [KS18].	23
3.6	Ressourcenverbrauch der Operation Addition auf einem FPGA [KS18].	24
3.7	AND	30
3.8	OR	30
3.9	XOR	30
3.10	Simulationsparameter [KS18].	49
3.11	Simulationsparameter [KS18].	53
3.12	Varianz und Gruppenlaufzeit der Operation Multiplikation (Zweiter Eingang, falls unterschiedlich) [KS18].	55
3.13	Ressourcenverbrauch der Operation Multiplikation auf einem FPGA [KS18].	55
3.14	Varianz und Gruppenlaufzeit der Operation Begrenzung [KS18].	57
3.15	Ressourcenverbrauch der Operation Begrenzung auf einem FPGA [KS18].	58
4.1	Motordaten [6]	64
5.1	Ressourcenverbrauch der Operation Begrenzung auf einem FPGA [MTK18].	93

Symbolverzeichnis

Darstellung

x, X	Skalar
\underline{x}	Vektor
$\underline{\underline{X}}$	Matrix

Häufig verwendete Indizes

$I - IV$	Sektornummer
----------	--------------

Formelzeichen

Abkürzung	Beschreibung	Einheit
α_n	N. Bit des Integers α	
BS	Bitstrom	
BS_1	1. Eingangsbitstrom	
BS_2	2. Eingangsbitstrom	
BS_{aus}	Ausgangsbitstrom	
BS_{ein}	Eingangsbitstrom	
C	Konstanter Offset	
f	Allgemeine Funktion	
f_s	Abtastfrequenz	Hz
g	Anzahl erwarteter Einsen	
N	Länge des betrachteten Bitstrom(ausschnitt)s	
o	Offset	
p	Wahrscheinlichkeit einer Eins im Bitstrom	
t_d	Laufzeit	s
t_m	Messverzögerung	s

v	Signalinformation
V_{0_0}	Nullvektor mit allen Phasen negativ versorgt
V_{0_1}	Nullvektor mit allen Phasen positiv versorgt
V_1	Aktiver Vektor mit einer Phase positiv versorgt
V_2	Aktiver Vektor mit zwei Phasen positiv versorgt
w	Anzahl Einsen
z	Anzahl Nullen

Abkürzungen

ADU	Analog-zu-Digital Umsetzer
AND	Logische Operation Und
ANV	Auslassen von Nullvektoren
DAU	Digital-zu-Analog Umsetzer
$\Delta\Sigma$ -M	$\Delta\Sigma$ -Modulator
$\Delta\Sigma$ -PWM	$\Delta\Sigma$ Pulsweitenmodulation Modulator
$\Delta\Sigma$ SV	$\Delta\Sigma$ Signalverarbeitung
DSV	digitale Signalverarbeitung
EMK	Elektromagnetische Kraft
FPGA	Field Programmable Gate Array
IGBT	Bipolartransistor mit isolierter Gate-Elektrode
LSB	Bit mit dem niedrigsten Stellenwert
MSB	Bit mit dem höchsten Stellenwert
MOSFET	Metall-Oxid-Halbleiter-Feldeffekttransistor
NBS	“Nullbitstrom”
NOT	Logische Operation Nicht
OR	Logische Operation Oder
P	proprtional
PI	proprtional integral
PLL	Phasenregelschleife
PMSM	Permanentmagnet-Synchronmotor

PWM	Pulsweitenmodulation
SDRAM	synchroner dynamischer Speicher mit wahlfreiem Zugriff
SINAD	Signal-zu-Rausch- und Verzerrungsverhältnis
RZM	Raumzeigermodulation
ÜM	Übermodulation
ÜMÜ	Übermodulation mit Überlaufbegrenzung
XOR	Logische Operation exklusives Oder

Kurzfassung

In modernen Anwendungen werden überabtastende Analog-zu-Digital Umsetzer eingesetzt. Eine besondere Klasse sind die Delta-Sigma-Analog-zu-Digital Umsetzer (ADU). Sie werden bei sehr hohen Abtastfrequenzen betrieben, in der Regel bei dem 64- bis 1024-fachen der Signalfrequenz, dafür besitzen sie eine sehr geringe Quantisierung, bis zu einem Bit. Mit entsprechenden digitalen Filtern lassen sich mit dem Verfahren Wortbreiten von 12 bis 16 Bit erreichen. Allerdings begrenzt dabei das Filter die erzielbare Kleinsignalbandbreite. Mit der weiterentwickelten und vorgestellten $\Delta\Sigma$ Signalverarbeitung ($\Delta\Sigma\text{SV}$) kann auf die Filterung verzichtet werden.

In dieser Arbeit werden unterschiedliche Verfahren zur $\Delta\Sigma\text{SV}$ evaluiert. Das Ergebnis stellen sowohl Klassen für lineare, als auch eine Klasse für nichtlineare Operationen dar, die eine gute Abbildungsqualität besitzen und alle Funktionen aus der jeweiligen Klasse abbilden können.

Basierend auf diesen Operationen wird die Strom- und Spannungsregelung einer dreiphasigen Last vorgestellt. Zur Umsetzung der hochfrequenten Bitströme in leistungshalbleitertaugliche Schaltfrequenzen wird ein hysteresebasierter Modulator, der echte Raumzeigermodulation mit beiden Nullvektoren beherrscht, eingesetzt. Für diesen wird ein Schaltfrequenzregler vorgestellt und unterschiedliche Stromregler erprobt.

Abschließend wird eine drehgeberlose Geschwindigkeitsregelung in $\Delta\Sigma\text{SV}$ vorgestellt.

Abstract

In modern applications oversampling analog-to-digital converters (ADC) are used. A special class are the Delta-Sigma-ADCs. They operate at very high sampling frequencies, usually at 64 up to 1024 times the signal frequency, but have a very low quantization, down to one bit. With corresponding digital filters, word sizes from 12 to 16 Bit can be achieved. However, the filters limit the achievable small signal bandwidth. In order to omit these, the $\Delta\Sigma$ signal processing ($\Delta\Sigma$ SP) has been developed.

In this thesis different methods for $\Delta\Sigma$ SP are evaluated. As a result, a class for linear as well as a class for nonlinear operations is presented, which have a good projection and can map all functions of the respective class.

Based on these operations, the current and voltage regulation of a three-phase load is presented. To convert the high-frequency bitstreams into switching frequencies suitable for power semiconductors, a hysteresis-based modulator is presented, which outputs true space vector modulation with both zero vectors. For this modulator a switching frequency controller is presented and different current controllers are evaluated.

Concluding, an encoderless speed control is presented in $\Delta\Sigma$ SP.

1 Einleitung

Die Regelungstechnik wurde bereits in der Antike für technische Regelungen eingesetzt, wie zum Beispiel bei der Wasseruhr des Ktesibios [38], um das Regelziel zu erreichen. Bei dem Beispiel wird der Wasserzulauf in das Schwimmergefäß dadurch reguliert, dass im vorgeschalteten Gefäß der Wasserstand durch einen Schwimmer konstant gehalten wird und somit immer mit dem gleichen statischen Druck auf den Ablauf drückt. Somit konnte der Gleichlauf der Uhr bei der Speisung aus einer unzuverlässigen Quelle verbessert werden. Seit den 1930er Jahren kamen einheitliche, einstellbare PID Regler auf. Aufgrund der fallenden Preise für Rechenleistung werden immer mehr Prozesse geregelt und neue Funktionen durch geeignete Regelungsverfahren ermöglicht. Genauso werden mit Hilfe der Regelungstechnik Materialkosten in der Herstellung gespart, da die geforderten Ergebnisse mit verbesserter Regelung mit günstigeren Maschinen und Bauteilen erreicht werden.

Heutzutage wird in der Messtechnik häufig auf überabtastende Analog-zu-Digital Wandler gesetzt, deren Quantisierung durch nachfolgende Filterung erhöht wird. Ein besonderes Beispiel hierfür sind $\Delta\Sigma$ -Modulatoren ($\Delta\Sigma$ -M). Diese Analog-zu-Digital Umsetzer (ADU) tasten das zu messende Signal hochfrequent ab (≥ 10 MHz) und quantisieren mit 1-5 bit. Mit Hilfe von Tiefpassfiltern werden Auflösungen ≥ 14 bit bei Abtastfrequenzen ≥ 16 kHz erreicht. In den vorgestellten Untersuchungen wird immer von einem $\Delta\Sigma$ -M mit 1 bit Quantisierung ausgegangen.

1.1 Stand der Forschung

In dieser Arbeit werden unterschiedliche Verfahren zur $\Delta\Sigma$ Signalverarbeitung ($\Delta\Sigma$ SV) vorgestellt und weiterentwickelt. Die zu messenden Analogwerte werden mit günstigen $\Delta\Sigma$ -M gemessen und die Datenverarbeitung

findet im Abtastungstakt ($\geq 10\text{ MHz}$) der Modulatoren statt. Dabei ist zu berücksichtigen, dass das Ausgangssignal eines $\Delta\Sigma$ -M keinen Augenblickswert besitzt, da der $\Delta\Sigma$ -M nur ein Bit pro Takt ausgibt. Im Gegensatz zur klassischen Signalverarbeitung, für die ein Multibit-Datenwort durch Tiefpassfilterung der Bitströme ermittelt wird, verarbeiten die $\Delta\Sigma\text{SV}$ Verfahren direkt die hochfrequenten Bitströme.

Erste Ansätze finden sich dazu in [10, 13, 17–19, 29, 30]. Diese Ansätze wurden in [KS18] bewertet, weiterentwickelt und um weitere nichtlineare Operationen erweitert. Mit den vorgestellten Operationen lassen sich alle Regler und Beobachter implementieren, die begrenzte Eingänge, Ausgänge und interne Zustände haben.

Aufgrund der Tatsache, dass die Ausgangssignale hochfrequente Bitströme ($\geq 10\text{ MHz}$) sind, wird eine Pulsweitenmodulation (PWM) benötigt, die die Schaltfrequenz auf leistungshalbleitertaugliche Frequenzen ($\leq 40\text{ kHz}$) abbildet. Für drei-phasige Lasten gibt es mehrere Lösungen [4, 5, 10–12]. Dabei hebt sich der $\Delta\Sigma$ Pulsweitenmodulation Modulator ($\Delta\Sigma$ -PWM) nach [10, 11] von den anderen Implementierungen ab, da er die Nullvektoren wie bei der klassischen Raumzeigermodulation verwendet und somit gleiche Schaltfrequenzen auf allen Phasen erreicht. In dieser Arbeit wird noch einmal das Grundprinzip und die erreichten Verbesserungen des $\Delta\Sigma$ -PWM dargestellt. Dies sind insbesondere die online Erkennung des aktuellen Betriebsmodus ohne zusätzliche Parametrisierung, die Auslegung für den Betragsabstand der Erkennung von Integratorüberläufen zur Schalthysterese, das Vereinfachen der Betragshysterese und die Schaltfrequenzregelung.

1.2 Zielsetzung der Arbeit

Die aus der Literatur bekannten Verfahren zur $\Delta\Sigma\text{SV}$ sollen weiterentwickelt und verglichen werden, sowie für die Modellierung linearer Systeme vergleichbar bewertet werden. Weiter sollen neue Operationen eingeführt werden, die nichtlineare Funktionen auf $\Delta\Sigma$ -Bitströmen ermöglichen. Das Ziel ist es, eine Operationsklasse zu erarbeiten, die beliebige nichtlineare Operationen darstellen kann. Um die Ergebnisse der $\Delta\Sigma\text{SV}$ Operationen

an einer dreiphasigen Last zur Wirkung zu bringen, soll der am Institut für Regelungstechnik von Herrn Homann entwickelte $\Delta\Sigma$ -PWM weiterentwickelt werden. Diese Fortschritte sollen abschließend in Anwendungen der elektrischen Antriebsregelung evaluiert werden.

2 Delta-Sigma Modulator

Die $\Delta\Sigma$ SV basiert auf der Veränderung eines sogenannten Bitstroms. In diesem Kapitel wird erklärt, wie ein sogenannter Bitstrom entsteht und welche charakteristischen Eigenschaften er besitzt.

In der klassischen Implementierung wird ein $\Delta\Sigma$ -ADU aus zwei Elementen, dem $\Delta\Sigma$ -M und einem Tiefpassfilter mit Dezimierung, wie in Abb. 2.1 gezeigt aufgebaut. Durch das Weglassen des Tiefpassfilters und der direkten Signalverarbeitung des hochfrequenten Bitstroms ($\geq 10\text{ Mbit/s}$) [10,12,13], [HKKS16,HKS16] wird die Gruppenlaufzeit des Tiefpassfilters im Regelkreis eingespart und somit eine größere Kleinsignalbandbreite erreicht.

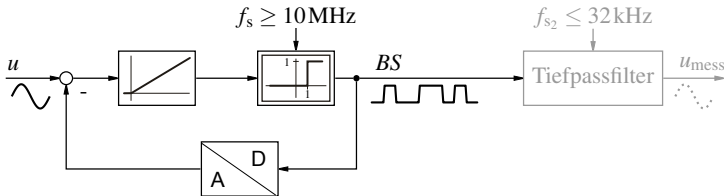


Abbildung 2.1: $\Delta\Sigma$ -M erster Ordnung mit Tiefpassfilter [KTS18a]

Der $\Delta\Sigma$ -M besteht aus einem oder mehreren Integratoren und einem getakteten Komparator mit Rückkopplungsschleifen und Verstärkungen. Die Ordnung des $\Delta\Sigma$ -M wird durch die Anzahl der implementierten Integratoren bestimmt, d.h. in Abb. 2.1 wird ein $\Delta\Sigma$ -M erster Ordnung gezeigt. Während ein $\Delta\Sigma$ -M erster Ordnung immer stabil ist, müssen die Rückkopplungskoeffizienten bei $\Delta\Sigma$ -M höherer Ordnung zu den einzelnen Stufen für Stabilität ausgelegt werden [34]. Aus der Literatur sind mehrere Entwurfsmethoden bekannt [25,33]. Im Folgenden wird, sofern nicht explizit genannt, von $\Delta\Sigma$ -M zweiter Ordnung ausgegangen.

Die Funktion lässt sich am Besten an einem $\Delta\Sigma$ -M 1. Ordnung mit einem 1-

Bit Quantisierer nachvollziehen. Als geschlossener Regelkreis integriert der $\Delta\Sigma$ -M den Regelfehler zwischen dem Eingangssignal und dem Ausgangssignal. Ist dieses Integral größer als Null, gibt der getaktete Quantisierer eine Eins aus, ansonsten eine Null. Zur Bildung des Regelfehlers wird der Bitstrom von einem 1-Bit Digital-zu-Analog Umsetzer (DAU) transformiert. Dieser gibt entweder die maximal, oder die minimal messbare elektrische Größe aus. Dieses einfache Regelgesetz stellt in Verbindung mit der Rückführung sicher, dass der Ausgangsbitstrom nur aus gleichgewichteten Einsen und Nullen besteht, die das Eingangssignal im Mittel darstellen. Das gilt für alle möglichen Fensterlängen, wobei die Messung genauer wird, je länger der betrachtete Ausschnitt ist.

Durch die Quantisierung mit zwei Zuständen wird dem Nutzsignal bei jeder Abtastung ein energiereiches Fehlersignal hinzugefügt [34]. Aufgrund der inneren Struktur erzeugt der $\Delta\Sigma$ -M eine Rauschverformung für das Quantisierungsrauschen. Wie in Abb. 2.2 gezeigt, wird die Rauschenergie hauptsächlich in die hochfrequenten Signalkomponenten verschoben. In einer klassischen digitalen Signalverarbeitung (DSP) Implementierung werden diese durch ein digitales Tiefpassfilter unterdrückt. In $\Delta\Sigma$ SV bleibt dieses Rauschen in den Bitströmen und beeinflusst dadurch die lokale Verteilung von Einsen und Nullen.

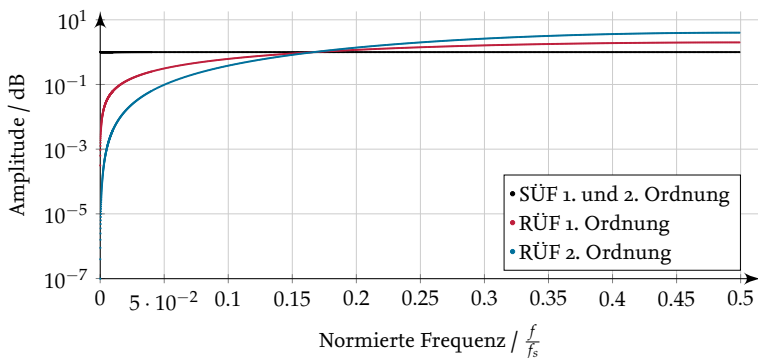


Abbildung 2.2: Betrag der Signalübertragungsfunktion (SÜF) und der Rauschübertragungsfunktion (RÜF) eines $\Delta\Sigma$ -M erster Ordnung und eines $\Delta\Sigma$ -M zweiter Ordnung [HKKS16].

Die Ordnung eines $\Delta\Sigma$ -M ergibt sich aus der Anzahl der in Reihe geschalteten Integratoren im Signalpfad. Wie bereits in Abb. 2.2 gezeigt, erreichen $\Delta\Sigma$ -M höherer Ordnung eine bessere Rauschverformung und können somit ein besseres Signal-zu-Rausch- und Verzerrungsverhältnis (SINAD) erreichen [34]. Dies liegt unter Anderem an der Vermeidung von Ruhetönen. Damit sind wiederkehrende Bitfolgen gemeint, die durch ihre Sequenzlängen Frequenzen im Nutzsignalband erzeugen. Diese treten insbesondere bei anliegenden Gleichgrößen auf [7]. $\Delta\Sigma$ -M höherer Ordnung neigen zur Instabilität und müssen durch Parametrisierung stabilisiert werden [25, 34], ein Tool zur Auslegung ist [33].

Eine allgemeine Darstellung nach [34] eines $\Delta\Sigma$ -M zweiter Ordnung mit Nullstellen und optionaler resonanter Rückführung ist in Abb. 2.3 dargestellt.

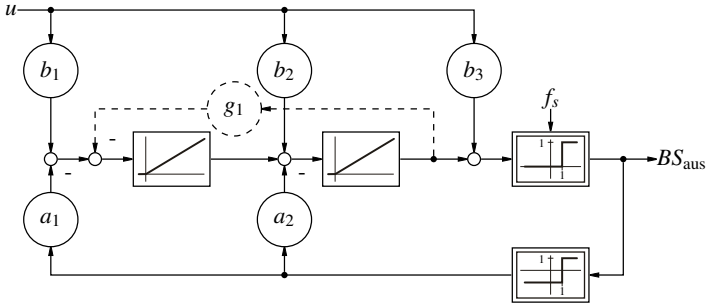


Abbildung 2.3: Allgemeine Darstellung eines $\Delta\Sigma$ -M zweiter Ordnung nach [34].

Die $\Delta\Sigma$ -M können auf zwei Arten implementiert werden. In einer Hardwareimplementierung ist der Integrator eine analoge Schaltung und die Rückkopplung ein Ein-Bit-Digital-Analog-Umsetzer, während bei einer Softwareimplementierung der gesamte Modulator digital implementiert ist.

Generell gibt es zwei Möglichkeiten, die logischen Werte *wahr* und *falsch* bzw. 1 und 0 auf normierte Signalwerte abzubilden: mit und ohne Vorzeichen. Im vorzeichenbehafteten Fall wird die logische 1 auf 1 und die

logische 0 auf -1 abgebildet, während im vorzeichenlosen Fall die logische 0 auf 0 abgebildet wird.

Im Bereich der Steuerungssysteme können Signale in der Regel positive und negative Werte haben, sodass die vorzeichenbehaftete Darstellung im Allgemeinen angenommen wird.

3 Delta-Sigma Signal Verarbeitung

Die im vorigen Kapitel vorgestellten $\Delta\Sigma$ -M erzeugen hochfrequente Bitströme. Diese bestehen in der Regel aus Nullen und Einsen und haben daher keinen Augenblickswert. Um Bitströme für Regelungssysteme zu nutzen, werden entweder Tiefpassfilter für klassische DSV oder eine neue Art der Datenverarbeitung, die $\Delta\Sigma$ SV verwendet, welche keine Tiefpassfilter benötigt und somit eine höhere Kleinsignalbandbreite ermöglicht. In diesem Kapitel werden verschiedene Implementierungsklassen von $\Delta\Sigma$ SV vorgestellt und bewertet. Die vorgestellten Lösungen lassen sich in zwei Arten unterteilen, lineare und nichtlineare Operationen. Aufgrund des Fehlens eines Augenblickswertes und des Quantisierungsrauschens im oberen Frequenzbereich sind nichtlineare Operationen ein besonderes Thema. In [KS18] werden lineare und nichtlineare Operationen dargestellt und bewertet. Die nachfolgenden Diskussionen basieren auf diesen Ergebnissen.

3.1 Lineare Bitstrom Operationen

In der Regel werden drei Verarbeitungsschritte durchgeführt, um eine mathematische Operation in $\Delta\Sigma$ SV durchzuführen. Im ersten Schritt wird der eingehende Bitstrom für die Operation vorbereitet, was zu einer Wortbreite von 1 bis k Bit führt, im nächsten Schritt wird die arithmetische Operation, in der Regel mit einem Multibit-Wert, durchgeführt. Im letzten Schritt wird das Multibit-Ergebnis in ein Ein-Bit-Signal, den Ausgangs-Bitstrom, umgewandelt.

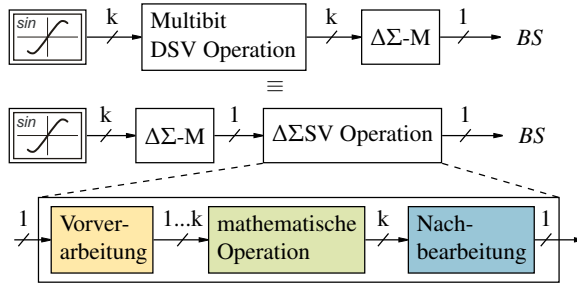


Abbildung 3.1: Grundsätzlicher Aufbau einer $\Delta\Sigma SV$ Operation mit den drei Verarbeitungsschritten und der äquivalenten DSV Operation [KS18].

Im Folgenden werden die Operationsklassen Logik Operationen [27], Delta Addierer [17], Bit Modifikation [10], Quantendekodierung [30], Erweiterte Quantendekodierung [KS18] und Bipolare Interpretation [KS18] vorgestellt und besprochen. Das Hauptproblem bei linearen Operationen ist, dass die im Bitstrom dargestellte Information (v) in Abhängigkeit vom dargestellten Signal (u) und einem festen Offset (o) enthalten sind. Dies kann wie folgt dargestellt werden

$$v + o = \frac{1}{2} u + \frac{1}{2}. \quad (3.1)$$

3.1.1 Logik Operation

Basierend auf der Tatsache, dass Bitströme die logischen Zustände *wahr* und *falsch* annehmen, können logische Operationen direkt Bitströme verarbeiten [Kle18]. Somit entfallen der Vorverarbeitungsschritt und der Nachbearbeitungsschritt. Die logische Operation Nicht (NOT), deren Wahrheitstabelle in Tabelle 3.1 dargestellt ist, führt zu einer Negation des vorzeichenbehafteten Eingangsbitstroms [27].

Tabelle 3.1: NOT

x	y
0	1
1	0

Jede Eins im Bitstrom wird von NOT in eine Null konvertiert und andersherum. Somit ergibt sich für eine gegebene Wahrscheinlichkeit von Einsen im Eingangsbitstrom, immer diese Wahrscheinlichkeit für Nullen im Ausgangsbitstrom, ohne dass Variationen auftreten können. Daraus folgt, dass die Varianz Null ist. Die mathematische Funktion, die den Mittelwert von NOT beschreibt, ist

$$f_{not}(x_1) = -x_1 : \forall x_1 \in [-1 \dots 1]. \quad (3.2)$$

Die logische Operation NOT eignet sich sehr gut um eine Negation durchzuführen, da das Rauschen des Bitstroms nicht verändert wird.

3.1.2 Delta Addierer

Die früheste Methode zur Realisierung von Berechnungen in der Bitstromebene basiert auf Volladdier- und Verzögerungslogik [17]. Wie in [HKKS16] gezeigt, ist die Operation äquivalent zu einer Addition mit Division durch zwei und anschließender Modulation mit einem $\Delta\Sigma$ -M erster Ordnung. Die Division durch zwei korrigiert den Offset, da die Hälfte von zwei Offsets von Bitströmen den korrekten Offset im Ausgangsbitstrom ergibt, mathematisch geschrieben

$$o = \frac{o_1 + o_2}{2} = \frac{\frac{1}{2} + \frac{1}{2}}{2} = \frac{1}{2}. \quad (3.3)$$

Die klassische Implementierung mit Logikbausteinen ist in Abb. 3.2 und das entsprechende regelungstechnische Blockschaltbild in Abb. 3.3 dargestellt.



Abbildung 3.2: Delta Addierer aus Logikbausteinen [17] und das vereinfachte Blockschaltbild [KS18].

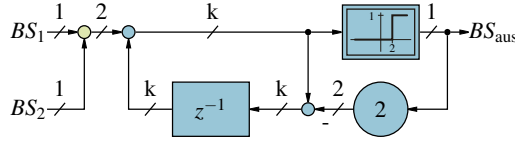


Abbildung 3.3: Äquivalentes regelungstechnisches Blockschaltbild eines Delta Addierers [HKKS16].

Basierend auf dieser Technologie wurde eine Multiplikation mit einem ganzzahligen Multibit-Wert α für Skalierungsfaktoren $0 \leq \alpha \leq 1$ vorgestellt [17]. Die Multiplikation basiert auf der inhärenten Division durch zwei eines Delta Addierers. Die Skalierung wird durch eine Kaskade von Delta Addierern erreicht, wobei ein Eingang mit einer Konstanten *wahr* oder *falsch* den Skalierungsfaktor in binärer Schreibweise darstellt, wie in Abb. 3.4 gezeigt.

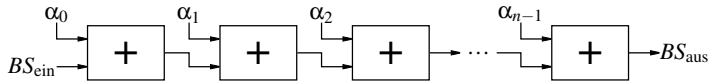


Abbildung 3.4: Skalierung basierend auf Delta Addierern, wobei α_i dem i . Bit des Skalierungsfaktors vom höchstwertigen Bit entspricht [17].

Die Anzahl der zu implementierenden Delta Addierer skaliert linear mit der Bitbreite (n) von α . Wobei α_0 das höchstwertige und α_{n-1} das niederwertige Bit des Skalierungsfaktors darstellt. Die resultierende Funktion kann im Mittel wie folgt ausgedrückt werden

$$\overline{BS}_{\text{aus}} = \overline{BS}_{\text{ein}} \cdot \sum_{i=1}^n \left[\alpha_{n-i} \cdot \left(\frac{1}{2} \right)^i \right]. \quad (3.4)$$

3.1.3 Bitstrommodifikation

Eine andere in der Literatur vorgestellte Technologie ist die Bitstrom-Modifikation [10]. Sie basiert auf der Idee, dass das Signal im Bitstrom aus zwei Teilen besteht, dem Signalteil und dem konstanten Offset. Die

Addition in der Bitstrom-Modifikation kann als Subtraktion des Festwertes 0,25, als Addition und als Speicher für überschüssige Einsen dargestellt werden. Diese werden erzeugt, wenn zwei Einsen gleichzeitig ausgegeben werden sollen. Dieser Speicher ist ein modifizierter $\Delta\Sigma$ -M erster Ordnung [HKKS16]. Um 0,25 zu subtrahieren, wird in jedem vierten Taktzyklus eine Eins in eine Null umgewandelt. Ist der Bitstrom zu dem Zeitpunkt gleich Null, wird die nächste Eins transformiert. Abb. 3.5 zeigt das detaillierte Blockschaltbild. Die Subtraktion des Festwertes 0,25 für beide Eingangsbitströme ergibt den korrekten Offset im Ausgangsbitstrom, wie aus folgender Rechnung ersichtlich wird

$$o = o_1 - 0,25 + o_2 - 0,25 = \frac{1}{2} - 0,25 + \frac{1}{2} - 0,25 = \frac{1}{2}. \quad (3.5)$$

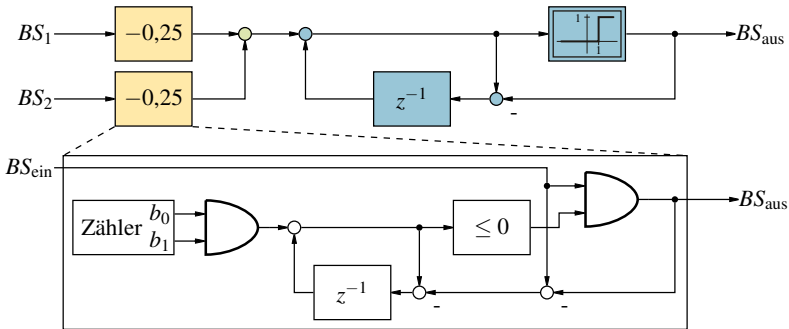


Abbildung 3.5: Addition zweier Bitströme mit Bitstrommodifikation und die detaillierte Darstellung von -0,25 [KS18].

Skalierungsoperationen können auch mittels Bitstrom-Modifikation durchgeführt werden. Zur Beibehaltung des festen Offsets muss eine zusätzliche Logik implementiert werden. Dabei wird zwischen Skalierungen größer und kleiner Eins unterschieden.

Bei einem Skalierungsfaktor größer Eins wird mit dem Signalanteil auch der feste Offset skaliert, somit muss dieser im Bitstrom reduziert werden. Ein Bitstrom kann keine normierten Werte größer Eins darstellen, somit muss ein Überlauf zu vermeiden werden. Daher sollte der konstante Offset vor der Skalierung reduziert werden, um Skalierungen größer Zwei für den

Eingangswert Null zu ermöglichen. Die Anzahl der gelöschten Einsen (o_d) lässt sich aus dem Skalierungsfaktor (s) und der Abschnittslänge (n) wie folgt ableiten

$$o_d = \left(\frac{1}{2} s - \frac{1}{2} \right) \frac{n}{s}. \quad (3.6)$$

Bei einem Skalierungsfaktor kleiner Eins wird der feste Offset verkleinert, sodass dem Bitstrom weitere Einsen hinzugefügt werden müssen. Um während des Betriebs keinen Überlauf zu erzeugen, sollten diese im Ausgangsbitstrom eingefügt werden. Die Anzahl der zusätzlichen Einsen (o_a) kann aus dem Skalierungsfaktor (s) und der Abschnittslänge (n) abgeleitet werden

$$o_a = \left(\frac{1}{2} - \frac{1}{2} s \right) n. \quad (3.7)$$

Mit dieser Information kann die minimale Abschnittslänge (n) berechnet werden, sodass eine natürliche Zahl für die geänderten Bits entsteht (o_x [steht für o_d oder o_a]). Eine gute Implementierung verteilt die geänderten Bits gleichmäßig in dem Abschnitt. Dies kann durch einen Überlaufzähler mit der Länge n und o_x Vergleichen erreicht werden. Die Implementierung zur Reduzierung der Anzahl von Einsen kann ähnlich wie -0,25 in Abb. 3.5 sein, nur mit einer modifizierten Zählerlänge (n) und n_a Vergleichen anstelle der logischen Operation Und (AND) am Zählerausgang. In Abb. 3.6 wird das angepasste Blockschaltbild gezeigt, um Einsen in den Bitstrom einzufügen. Der Vergleichsblock gibt eine Eins aus, wenn der Eingabewert in der Liste steht.

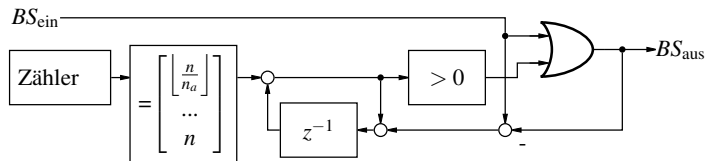


Abbildung 3.6: Addition eines festen Offsets zu einem Bitstrom [KS18].

3.1.4 Zählerbasiert

In [23] wird eine zählerbasierte Operationsklasse vorgestellt. Der Offset wird durch einen Zähler am Eingang entfernt, der die doppelte Summe der Eingänge bildet und die Anzahl an Eingängen abzieht, was für den Offset wie folgt beschrieben werden kann

$$o = 2 \cdot o_1 - 1 + 2 \cdot o_2 - 1 + o_M = \frac{2}{2} - 1 + \frac{2}{2} - 1 + \frac{1}{2} = \frac{1}{2}. \quad (3.8)$$

Der Vorbereitungsschritt führt zu dem gleichen Signal, wie die bipolare Interpretation im Abschnitt 3.1.7 auf Seite 19. Der Unterschied liegt in der Ausgangsstufe, die bei der zählerbasierten Operationsklasse von einem modifizierten $\Delta\Sigma$ -M gebildet wird. In dieser Arbeit werden der Addierer und die Kombination aus dem Addierer und dem Mittelwertaddierer zweiter Ordnung untersucht. Der Addierer erster Ordnung ist in Abb. 3.7 abgebildet.

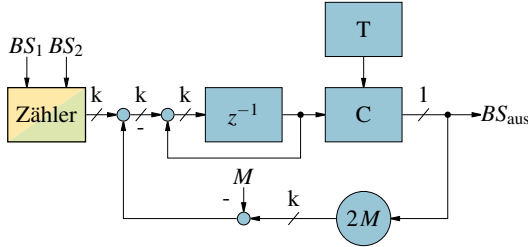


Abbildung 3.7: Zählerbasierte Addition erster Ordnung [KS18].

Der Addierer zweiter Ordnung wird aus dem vorgestellten Mittelwertaddierer zweiter Ordnung erzeugt, indem der Rückkopplungskoeffizient (M) auf 1 anstelle der Anzahl der Eingangsbitströme gesetzt wird. Dies führt zu der Topologie, die in Abb. 3.8 dargestellt ist.

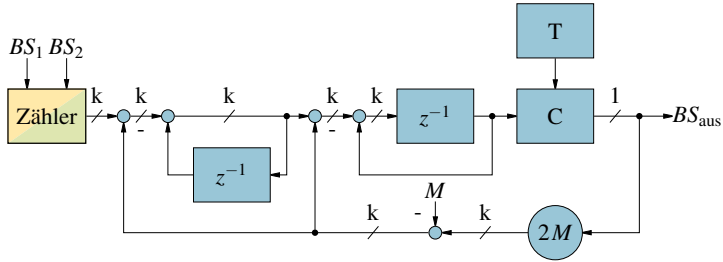


Abbildung 3.8: Zählerbasierte Addition zweiter Ordnung [KS18].

Die Modifikation des $\Delta\Sigma$ -M ist der implementierte Quantisierer. Für Eingangswerte ungleich Null entspricht das Verhalten dem des klassisch eingesetzten 1 bit Quantisierers. Bei Eingängen gleich Null unterscheidet sich der Quantisierer durch die zusätzliche Logik T. Diese bewirkt, dass abwechselnd Einsen und Nullen ausgegeben werden. Die Wahrheitstabelle des Quantisierers ist in Tabelle 3.2 angegeben.

Tabelle 3.2: Wahrheitstabelle des Quantisierers für zählerbasierte Operationen [KS18].

C_{ein}	C_{aus}	T_{neu}
< 0	0	T
> 0	1	T
0	T	NOT(T)

Basierend auf diesem Addierer ist ein Mittelwertaddierer in [23] präsentiert, der in der Rückführung einen Skalierungsfaktor M gleich der Anzahl der Eingänge besitzt. Der Mittelwertaddierer kann verwendet werden, um ein Signal zu verkleinern. Wenn m der Eingänge mit dem Signal und $M - m$ Eingänge mit dem Bitstrom eines Nullsignals (“Nullbitstrom” (NBS), das in Abschnitt 3.1.5 auf der nächsten Seite vorgestellt wird) verbunden sind. Dadurch wird eine Skalierung um m/M erreicht. Aus Implementierungsgründen ist $m, M \in \mathbb{N}$ erforderlich. Wenn die in [23] postulierte Bedingung $m < M$ aufgehoben wird, können Skalierungsfaktoren größer und kleiner Eins implementiert werden. Für das Hochskalieren müssen Vorkehrungen

getroffen werden, um den darstellbaren Bereich des Bitstroms nicht zu überschreiten.

3.1.5 Quantendekodierung

Die Operationsklasse mit dem höchsten SINAD, die aus der Literatur bekannt ist, ist die Quantendekodierung [29,30], [HKKS16]. Für diese Operationsklasse wird ein sogenannter NBS verwendet, um den konstanten Offset zu entfernen. Der NBS wird durch einen vorzeichenbehafteten $\Delta\Sigma$ -M erster Ordnung mit dem Eingangssignal Null erzeugt. Daraus ergibt sich ein Bitstrom aus abwechselnden Einsen und Nullen. Dies entspricht einem getakteten Schalter, z.B. Abb. 3.9.

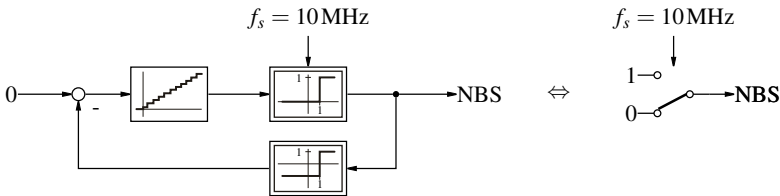


Abbildung 3.9: Nullbitstrom Generator basierend auf einem $\Delta\Sigma$ -M erster Ordnung und die vereinfachte Implementierung [KS18].

Der NBS wird von jedem Eingangssignal subtrahiert, was die Wahrheitstabelle Tabelle 3.3 implementiert. Die Quanten ($\pm Q$) werden als ± 1 interpretiert, was zu einem ternären Signal führt.

Tabelle 3.3: Wahrheitstabelle eines Bitstroms (BS) minus den NBS ergibt das quantendekodierte Signal (QD) [29].

NBS	BS	QD
0	0	0
0	1	+Q
1	0	-Q
1	1	0

Um den konstanten Offset nach der arithmetischen Operation wiederherzustellen, wird das Ergebnis mit einem digitalen bipolaren $\Delta\Sigma$ -M moduliert. Für die Operation Addition ergibt sich der Offset wie folgt

$$o = o_1 - o_{\text{ZBS}} + o_2 - o_{\text{ZBS}} + o_{\text{M}} = \frac{1}{2} - \frac{1}{2} + \frac{1}{2} - \frac{1}{2} + \frac{1}{2} = \frac{1}{2}. \quad (3.9)$$

Die Division durch Zwei, dargestellt in Gleichung (3.1), kann durch eine Skalierung mit dem Faktor Zwei kompensiert werden.

Zum Beispiel wird die in Quantendekodierung implementierte Operation Addition in Abb. 3.10 gezeigt.

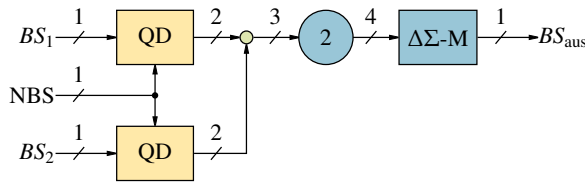


Abbildung 3.10: Die $\Delta\Sigma$ SV Operation Addition in Quantendekodierung [KS18].

Da der konstante Offset im ersten Schritt der Operation aus dem Bitstrom entfernt und im letzten Schritt wieder hinzugefügt wird, können Skalierungsoperationen ohne zusätzliche Vorsichtsmaßnahmen durchgeführt werden.

Der Nachteil dieser Lösung ist das Rauschen, den die sogenannten NBS verursachen. Dieser kann reduziert werden, indem man einen $\Delta\Sigma$ -M höherer Ordnung verwendet, um den NBS zu erzeugen. Diese Lösung führt zu erhöhten Implementierungskosten in logischen Elementen für den zusätzlichen $\Delta\Sigma$ -M.

3.1.6 Erweiterte Quantendekodierung

Eine verbesserte Version der Quantendekodierung ist die Erweiterte Quantendekodierung. Diese Operationsklasse verzichtet auf den NBS und damit das zusätzliche Rauschen. Der Nachteil dieser Operationsklasse ist, dass sie nur implementiert werden kann, wenn die lineare algebraische Operation

als Differenz einer geraden Anzahl von Signalen als erste Berechnung ausgedrückt werden kann. Der resultierende Bitstrom ist gleich dem Bitstrom der klassischen Quantendekodierung, wenn beide Eingangssignale zuerst mit dem gleichen NBS dekodiert werden und anschließend die Differenz berechnet wird. In Tabelle 3.4 ist die Logiktablelle für zwei NBS und zwei Eingangsbitströme (BS) als Operationseingang dargestellt. In den nächsten Spalten werden die Quantendekodierten Werte (QD) der Bitströme und die entsprechenden NBS angezeigt. In den letzten Spalten werden die Ergebnisse der algebraischen Operationen Subtraktion (-) und Addition (+) für die dekodierten Quantenwerte, wie die klassische Quantendekodierung und die direkt verarbeiteten Bitströme angezeigt. Wie zu sehen ist, sind in den hervorgehobenen Zeilen, in denen die verwendeten NBS gleich sind, die Ergebnisse für die algebraische Operation Minus mit und ohne Quantendekodierung gleich.

Tabelle 3.4: Wahrheitstabelle zweier Bitströme (BS_x) und deren “Nullbitströmen” (ZBS_x), die quantendekodierten Werte (QD_x), sowie die Ergebnisse von Subtraktion und Addition der quantendekodierten Werte als auch der direkt verarbeiteten Bitströme. In den hervorgehobenen Zeilen sind die “Nullbitströme” identisch [KS18].

ZBS_1	BS_1	ZBS_2	BS_2	QD_1	QD_2	-	+
0	0	0	0	0	0	0	0
0	0	0	1	0	1	-1	1
0	0	1	0	0	-1	1	-1
0	0	1	1	0	0	0	0
0	1	0	0	1	0	1	1
0	1	0	1	1	1	0	2
0	1	1	0	1	-1	2	0
0	1	1	1	1	0	1	1
1	0	0	0	-1	0	-1	-1
1	0	0	1	-1	1	-2	0
1	0	1	0	-1	-1	0	-2
1	0	1	1	-1	0	-1	-1
1	1	0	0	0	0	0	0
1	1	0	1	0	1	-1	1
1	1	1	0	0	-1	1	-1
1	1	1	1	0	0	0	0

Bei der algebraischen Operation Addition unterscheiden sich die Ergebnisse mit und ohne Quantendekodierung in einigen Fällen. Mit der Tatsache, dass NOT zu einer Invertierung des Vorzeichens eines bipolaren Bitstroms [27] führt, kann jede Addition auf eine Subtraktion abgebildet werden, wie in Abb. 3.11 gezeigt. In Regelkreisen wird meistens der Regelfehler als erste Operation berechnet, daher ist die Erweiterte Quantendekodierung ein guter Ansatz für $\Delta\Sigma\text{SV}$ in Regelungsanwendungen. Für eine ungerade Anzahl von Eingangssignalen kann ein NBS als Eingang hinzugefügt werden. Dies führt zu einer Kombination aus erweiterter und klassischer Quantendekodierung.

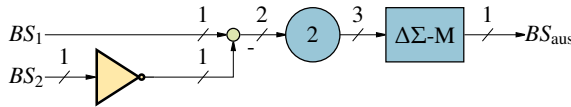


Abbildung 3.11: Die $\Delta\Sigma\text{SV}$ Operation Addition implementiert mit erweiterter Quantendekodierung [KS18].

3.1.7 Bipolare Interpretation

Für alle Fälle, in denen $\Delta\Sigma\text{SV}$ mit einer ungeraden Anzahl von Signalen in der ersten Operation verwendet werden soll, ist eine neue Operationsklasse zu entwickeln. Das Problem der Bitstrommodifikation und Quantendekodierung ist der Offset in den bipolaren Bitströmen. Dieser Offset kann auf Null reduziert werden, wenn die bipolare Interpretation des Bitstroms vor einer algebraischen Operation durchgeführt wird. Dazu wird eine Eins im Bitstrom auf den Wert eins und eine Null im Bitstrom auf den Wert minus eins abgebildet, bevor die Operation angewendet wird. Die grundsätzliche Funktionsweise ist in Abb. 3.12 dargestellt. Diese mathematische Definition eliminiert den parasitären Offset. Der große Vorteil ist, dass diese Operationsklasse nicht auf eine gerade Anzahl von Eingangssignalen oder zusätzlichen Nullsignalen, wie die NBS, beschränkt ist.

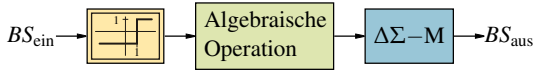


Abbildung 3.12: Basisimplementierung einer $\Delta\Sigma$ SV Operation in Bipolarer Interpretation [KS18].

Im Vergleich zu Quantendekodierung entfällt auch die inhärente Division durch Zwei. Der Eingangs- und Ausgangsbitstrom kann ein Bitstromvektor sein. Bei einem Ausgabe-Bitstromvektor werden mehrere $\Delta\Sigma$ -M eingesetzt. Im Falle eines Eingangsvektors wird jeder Bitstrom einzeln interpretiert, wie in Abb. 3.13 für die algebraische Operation Addition gezeigt.

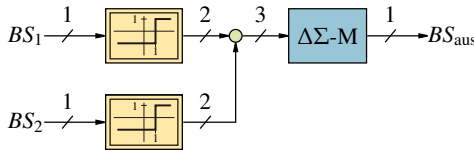


Abbildung 3.13: Die $\Delta\Sigma$ SV Operation Addition implementiert in Bipolarer Interpretation [KS18].

3.1.8 Evaluation

Der Bitstrom besteht nur aus Einsen und Nullen und daher ist es nicht möglich, ihm einen augenblicklichen Wert zuzuordnen. Daher werden die Bitströme aufgezeichnet und offline ausgewertet. Die Bitströme aus Simulationen können mit Hilfe von zweiseitigen Sinc-Filtern oder idealen Tiefpassfiltern ausgewertet werden. Für die Ergebnisse der Echtzeit-Implementierungen stehen nur einseitige Sinc-Filter zur Verfügung.

Für lineare Operationen sind der SINAD und die Gruppenlaufzeit die zentralen Kenngrößen zur Messung der Abbildungsqualität [HKKS16]. Der SINAD gemäß Gleichung (3.10) wird aus dem Signal (S), dem Rauschen (N) und der Verzerrung (D) berechnet und ist mit der effektiven Anzahl der Bits (ENOB) verknüpft, wie in Gleichung (3.11) dargestellt [14]. Das macht den SINAD zu einem guten Metrum, um die Informationen im Bitstrom zu quantifizieren.

$$\text{SINAD} = 20 \cdot \log \left(\frac{S}{N + D} \right) \quad (3.10)$$

$$\text{ENOB} = \frac{\text{SINAD} - 1,76 \text{ dB}}{6,02 \text{ dB}} \quad (3.11)$$

Zur Ermittlung der dynamischen und Abbildungseigenschaften der Operationen werden eine Sinusanregung und eine Sprunganregung als Eingangssignal verwendet. Als Referenz dient eine Multibit-DSV Operation deren Ergebnis $\Delta\Sigma$ moduliert wird. Der Signalverlauf für den Vergleich ist in Abb. 3.14 gezeigt. Die Gruppenlaufzeit ergibt sich aus der Zeitdifferenz der Sprungantworten der implementierten Operation in $\Delta\Sigma\text{SV}$ und der Referenz, die beide mit dem gleichen Tiefpassfilter ausgewertet werden. Beide Signale gelten als eingeschwungen, wenn sie 90 % der Eingangsamplitude erreicht haben.

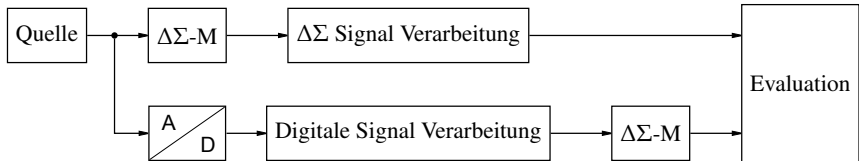


Abbildung 3.14: Signallaufplan für die Evaluation von $\Delta\Sigma\text{SV}$ Operationen [KS18].

3.1.9 Ergebnisse

Die vorgestellten $\Delta\Sigma\text{SV}$ Implementierungen werden mit dem im vorherigen Unterabschnitt beschriebenen Versuchsaufbau anhand der Operation Addition bewertet. Die beiden Eingangssignale werden von einer sinusförmigen Quelle mit den Amplituden 0,5 und 0,25 erzeugt. Die Sinuskurve hat eine feste Frequenz von 128 Hz bei einer Abtastfrequenz von 10 MHz. Um den SINAD als Maß für die Signalqualität zu berechnen, werden die Spektren der Bitströme mit Hilfe von idealen Tiefpassfiltern mit einer Dezimierungsrate von 32 berechnet. Die Ergebnisse sind in Abb. 3.15 dargestellt. Aufgrund der Tatsache, dass mit der logischen Operation NOT lediglich eine Negierung des Signals erreicht werden kann, kann sie

in diesem Vergleich nicht berücksichtigt werden. Allerdings verändert sie das Spektrum des Signals nicht, sodass sich der SINAD gegenüber dem Eingangssignal nicht verändert.

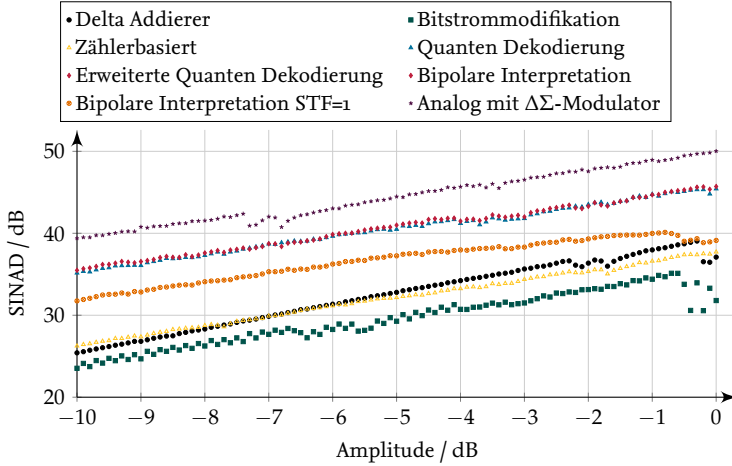


Abbildung 3.15: Signalqualität in SINAD nach unterschiedlichen $\Delta\Sigma$ SV Implementierungen einer Sinusschwingung mit variabler Amplitude [KS18].

Der resultierende SINAD der zählerbasierten Addition zweiter Ordnung ist nicht dargestellt, da er in allen Betriebspunkten negativ ist. Dieses Verhalten ist zu erwarten, da $\Delta\Sigma$ -M zweiter Ordnung für Stabilität ein Parameter Design erfordern und einheitliche Rückführungen nicht zu der Klasse der stabilen Auslegungen gehören [25, 34]. Somit wird nur die zählerbasierte Addition erster Ordnung diskutiert und als Zählerbasiert bezeichnet. Die beste Signalqualität wird mit einer analogen Operation und der anschließenden $\Delta\Sigma$ -Modulation erzielt. Da alle Operationen auf dem Prüfstand in $\Delta\Sigma$ SV zwei $\Delta\Sigma$ -M verwenden, einen für die Signalumwandlung des Multibit-Signals in einen Bitstrom und einen internen als Ausgangsstufe, während im Falle der DSV nur ein $\Delta\Sigma$ -M als Ausgangsstufe verwendet wird, ist dieses Ergebnis zu erwarten. Folglich bildet die Multibit-Operation gefolgt von einem $\Delta\Sigma$ -M die obere Grenze für die erreichbare Signalqualität. Die schlechteste Signalqualität erreichen

die Implementierungen basierend auf Delta Addierern, Bitstrommodifikation und Zählerbasiert. Dies liegt an den intern verwendeten $\Delta\Sigma$ -M erster Ordnung. Daraus folgt eine schwächere Rauschverformung des Quantisierungsrauschens, wie in Abb. 2.2 auf Seite 5 dargestellt. Die Implementierung Quantendekodierung erreicht im Durchschnitt eine um 0,20 dB geringere Signalqualität als Erweiterte Quantendekodierung. Für den Fall, dass nur eine Operation das gesamte Regelgesetz implementiert, ist dies vernachlässigbar. Bei kaskadierten Operationen akkumuliert sich das Rauschen mit jeder Stufe, sodass der Unterschied größer wird. Die Bipolare Interpretation zeigt die gleiche Performance wie die Erweiterte Quantendekodierung, da es eine gerade Anzahl von Eingangssignalen und keine vorherige Skalierung gibt, sodass beide Betriebsklassen den gleichen Bitstrom ergeben. In Anbetracht der Tatsache, dass die Skalierung eines einzelnen Eingangs nicht mit Erweiterter Quantendekodierung realisiert werden kann, wird die Bipolare Interpretation, vor allem aufgrund der Signalqualität, als die beste lineare Operationsklasse bewertet.

Tabelle 3.5: Gruppenlaufzeit der Operationsklassen in Taktzyklen [KS18].

Operationsklasse	Gruppenlaufzeit
Logik Operation	0
Delta Addierer	0
Bitstrommodifikation	1
Zählerbasiert	2
Quantendekodierung	9
Erweiterte Quantendekodierung	9
Bipolare Interpretation	9
Bipolare Interpretation STF = 1	0

Die zweite wichtige Kennzahl ist die Signallaufzeit. Die Ergebnisse in Tabelle 3.5 zeigen zwei Gruppen. Eine Gruppe hat fast keine Gruppenlaufzeit mit null bis einem Taktzyklus und die andere hat neun Taktzyklen Gruppenlaufzeit. Die Implementierungen basierend auf Delta Addierern und Bitstrommodifikation haben einen internen $\Delta\Sigma$ -M erster Ordnung und die Logik Operation kann komplett ohne zusätzlichen $\Delta\Sigma$ -M implementiert werden. Wie in [HKKS16] angegeben, hängt die Hauptverzögerungszeit

für die Quantendekodierung, und damit auch für die bipolare Interpretation, von dem verwendeten $\Delta\Sigma$ -M ab. In den Simulationen wird das Design mit minimalen Parametern aus [25] verwendet. Mit anderen Strukturen und Auslegungen, z.B. mit [33], ist eine verzögerungsfreie Signalübertragung möglich (STF=1). Der Nachteil dieser Lösung ist eine reduzierte Signalqualität. So kann ein Kompromiss zwischen Signallaufzeit und Signalqualität auf die jeweilige Anwendung zugeschnitten werden. Andererseits ist eine Gruppenlaufzeit von neun Taktzyklen bei ≥ 10 MHz ziemlich schnell für Systeme, die normalerweise bei ≤ 32 kHz geregelt werden.

Eine weitere wichtige Kennzahl ist der Ressourcenbedarf, welcher in Tabelle 3.6 dargestellt ist. Die Implementierungen, die auf Strukturen erster Ordnung basieren, benötigen in der Regel weniger Logikgatter als Implementierungen, die auf Strukturen zweiter Ordnung basieren. Beispielsweise verwendet die Implementierung Delta Adder nur sechs Logikgatter, während die Implementierungen für Strukturen zweiter Ordnung 28 bis 31 Logikgatter erfordern. Die Ausnahme ist die Bitstrommodifikation, die ebenfalls 20 Logikgatter erfordert. Dies ist auf die komplexe Subtraktion von $-0,25$ als Vorbereitungsschritt zurückzuführen. Basierend auf diesen Ergebnissen begünstigen die Implementierungskosten keine Implementierungsklasse zweiter Ordnung.

Tabelle 3.6: Ressourcenverbrauch der Operation Addition auf einem FPGA [KS18].

Implementierung	Logikgatter
Delta Addierer	6
Bitstrommodifikation	29
Zählerbasiert	17
Quanten Dekodierung	31
Erweiterte Quanten Dekodierung	28
Bipolare Interpretation	28
Bipolare Interpretation STF = 1	31

Die in Abb. 3.16 gezeigte Operation Integration bedarf besonderer Betrachtung. Der aktuelle Integratorwert, vom Typ Multibit wird zum neuen Eingangswert addiert, somit müssen beide Werte ohne Offset sein. Damit entfallen die Betriebsklassen Delta Addierer und Bitstrommodifikation. Theoretisch kann die Betriebsklasse Bitstrommodifikation durch Subtraktion von 0,5 statt 0,25 erweitert werden. Der Nachteil dieser Lösung ist, dass keine negativen Zahlen als Eingangssignal darstellbar sind. Für die Integration sollte eine Multibit-Begrenzung für den integrierten Wert eingeführt werden, die eine Übersteuerung des $\Delta\Sigma$ -M verhindert.

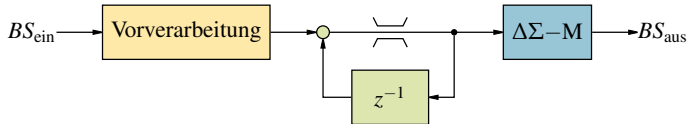


Abbildung 3.16: Allgemeine Darstellung der Integration in $\Delta\Sigma$ SV [KS18].

Mit den vorgestellten Operationsklassen Quantendekodierung, Erweiterte Quantendekodierung und Bipolare Interpretation können alle linearen Systeme und Regler mit begrenzten Eingängen, Ausgängen und internen Zuständen abgebildet werden.

3.2 Nichtlineare Bitstrom Operationen

Nichtlineare Funktionen in der $\Delta\Sigma$ Bitstrom Ebene müssen gesondert betrachtet werden. Die Hauptprobleme für nichtlineare Funktionen mit $\Delta\Sigma$ Bitströmen sind das Fehlen eines Augenblickswerts und die Veränderungen des Frequenzspektrums. Ohne Augenblickswert kann, zum Beispiel, nicht eindeutig erkannt werden, ob ein Grenzwert überschritten ist, sowie die Abbildung einer Wurzelfunktion mit Nullen und Einsen als Eingangssignal entspricht am Ausgang der Skalierung mit Eins. Des Weiteren können die Änderungen im Frequenzspektrum dazu führen, dass die von den $\Delta\Sigma$ -M eingeführte Rauschverformung, wie in Abb. 2.2 auf Seite 5 dargestellt, aufgehoben wird.

Basierend auf [KS18] wird eine allgemeine Klasse für nichtlineare Funk-

tionen vorgestellt und anhand der beiden Operationen Multiplikation und Begrenzung mit spezialisierten Implementierungen verglichen, die jeweils nur eine Operation abbilden können. Zunächst wird die allgemeine Klasse für nichtlineare Funktionen vorgestellt.

3.2.1 Schnelle Filter

Ein genereller Ansatz für nichtlineare Funktionen ist die Verwendung von schnellen Sinc-Filtern um aus den Bitströmen einen Multibit-Wert zu erzeugen. Schnell bedeutet in dem Fall, dass eine Gruppenlaufzeit im Bereich der linearen Operationsklassen erreicht wird, siehe Tabelle 3.5 auf Seite 23. Da das Ergebnis der arithmetischen Operation moduliert werden muss, wird die Gruppenlaufzeit des implementierten $\Delta\Sigma$ -M addiert. Die grundlegende Implementierung ist in Abb. 3.17 dargestellt.

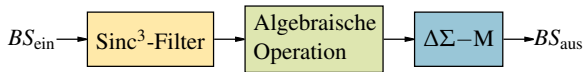


Abbildung 3.17: Allgemeine Implementierung einer nichtlinearen $\Delta\Sigma$ Funktion mit schnellem Filter [KS18].

Diese Klasse bietet mehrere Vorteile, die Gruppenlaufzeit ist konstant für alle Eingangsamplituden, jegliche nichtlineare Funktionen können mit klassischer DSV implementiert werden und durch das Tiefpassverhalten der Eingangsstufe wird das Quantisierungsrauschen des Bitstroms gedämpft. Insbesondere bei Multiplikationen, die einer Faltung im Frequenzbereich entsprechen, bewirkt die Filterung eine bessere Signalqualität, da das hochfrequente Quantisierungsrauschen, vergleiche Abb. 2.2 auf Seite 5, nur gedämpft in den Nutzfrequenzbereich gemischt wird.

Die Implementierung der Operation Multiplikation mit schnellen Filtern ist in Abb. 3.18 dargestellt.

Mit einem schnellen Filter ist auch die Implementierung einer Multibit-Limitierung möglich. Die resultierende regelungstechnische Struktur wird in Abb. 3.19 gezeigt.

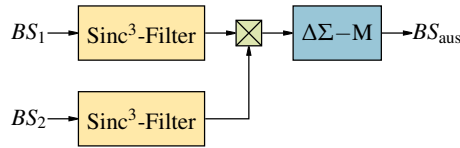


Abbildung 3.18: Multiplikation zweier Bitströme mit schnellen Filtern und DSV Operation [KS18].

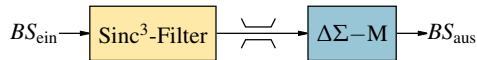


Abbildung 3.19: Limitierung eines Bitstroms mit schnellem Filter und DSV Operation [KS18].

3.2.2 Multiplikation

Bei der Multiplikation zweier Bitströme muss die Änderung des Frequenzspektrums berücksichtigt werden, denn im Frequenzbereich resultiert eine Faltung [16].

Zur Veranschaulichung der Effekte wird das Rauschspektrum eines $\Delta\Sigma$ -M mit einem voll besetzten Signalspektrum im Nutzfrequenzband überlagert. Dieses Signal wird als Beispiel gewählt, da es den größtmöglichen Signalinhalt hat. Die resultierenden Spektren für einen $\Delta\Sigma$ -M erster und einen zweiter Ordnung mit einem Überabtastungsverhältnis von 32 werden in Abb. 3.20 dargestellt. Aus diesen Spektren werden mögliche Signalfolgen im Zeitbereich mit der inversen schnellen Fourier-Transformation berechnet. In diesem Beispiel wird die Quadrierung als Repräsentant der Operation Multiplikation auf die Sequenzen angewendet. Im letzten Schritt werden die Spektren der resultierenden Sequenzen durch schnelle Fourier-Transformation (Spektrum quadriert in Abb. 3.20) ausgewertet. Der Signalverlauf verdeutlicht, dass das Rauschen in den quadrierten Signalen über den gesamten Frequenzbereich verteilt ist. Für die Signalkomponenten wird ein logarithmisches Dreieck mit der doppelten Breite und der maximalen Amplitude von 0 dB erwartet, da dies für die angenommenen Signale ohne Rauschen resultiert. Die zusätzliche Energie wird durch die Multiplikation in das Basisband gefaltet. Dies führt zu einer sehr schlechten Signalqualität, auch bei Verwendung eines Tiefpassfilters für das Ba-

sisband. Kurz gesagt, die Rauschformung wird durch Multiplikation ohne weitere Vorkehrungen aufgehoben.

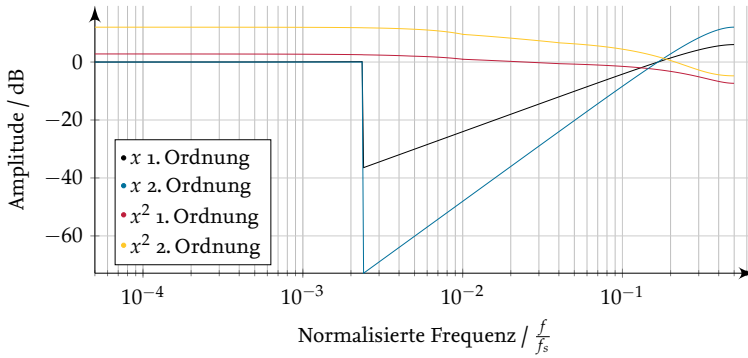


Abbildung 3.20: Ausgangsspektrum von $\Delta\Sigma$ -M erster und zweiter Ordnung und der im Zeitbereich quadrierten Signale [KS18].

Für die Operation Multiplikation werden drei spezialisierte Verfahren vorgestellt. Zwei Verfahren basierend auf Delta Addierern, Schieberegistern und Logikgattern und eine Implementierung basiert auf logischen Operationen.

3.2.2.1 Delta Addierer

Die ersten Ansätze zur Multiplikation zweier Bitströme werden in [21] vorgestellt. Zur Implementierung werden Logikgatter verwendet, sodass ein kurzes Filter entsteht und die Multiplikation berechnet wird. Der Nachteil dieser Methode ist die Anforderung, dass die Eingangsbitströme zueinander skalierte Abtastfrequenzen ($f_{s1} = n \cdot f_{s2}, n \in \mathbb{N}^*$) aufweisen müssen. Die dort vorgestellten Ergebnisse zeigen, dass ein höherer Faktor zwischen den Abtastraten zu einem höheren SINAD führt.

Eine verbesserte Version des Multiplikators kann zwei Bitströme mit gleicher Eingangsfrequenz verarbeiten [20]. Der verbesserte Multiplikator besteht aus einem 2^n Bit Schieberegister für den ersten Bitstrom und multipliziert jedes Register mit dem aktuellen Bit des zweiten Bitstroms. Die

Ergebnisse werden über eine Kaskade aus Delta Addierern gemittelt, wie in Abb. 3.21 dargestellt.

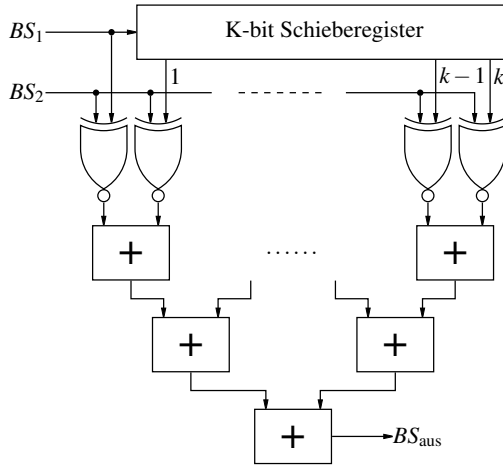


Abbildung 3.21: Multiplikation zweier Bitströme basierend auf Delta Addierern [20].

3.2.2.2 Delta Addierer Modifiziert

Durch die Rauschverformung des $\Delta\Sigma$ -M wird das Rauschen in den hochfrequenten Signalbereich verschoben und das k -Bit-Schieberegister wirkt zusammen mit den Delta Addierern wie ein Tiefpassfilter. Daher wird die Vermutung aufgestellt, dass ein zweites Schieberegister für den anderen Eingang eine weitere Verbesserung der Signalqualität bewirkt. Aufgrund der Symmetrie kann die logische Verknüpfung und die Speicherung im Schieberegister getauscht werden, was ein Schieberegister einspart. Die Implementierung ist in Abb. 3.22 dargestellt.

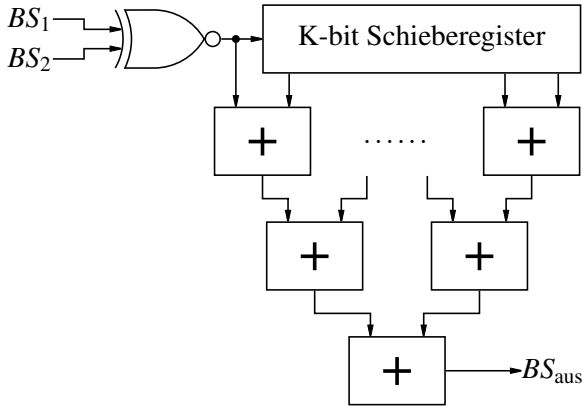


Abbildung 3.22: Modifizierte Multiplikation zweier Bitströme basierend auf Delta Addieren [KS18].

3.2.2.3 Logische Operationen

Wie bereits bei den linearen Funktionen beschrieben, können Bitströme mit einem Bit Information pro Taktzyklus direkt mit logischen Grundoperationen verarbeitet werden. In diesem Abschnitt werden die logischen Operationen mit zwei Eingängen AND, OR und XOR diskutiert. Diese Untersuchung ist in [Kle18] vorgestellt, auf dem dieser Abschnitt basiert. Die Wahrheitstabellen sind in Tabellen 3.7 bis 3.9 aufgeführt.

Tabelle 3.7: AND

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Tabelle 3.8: OR

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	1

Tabelle 3.9: XOR

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

3.2.2.3.1 Statistische Betrachtung Mit Hilfe statistischer Analysen wird ausgewertet, welche mathematischen Funktionen der Bitstrom

Mittelwertsänderung entsprechen, die durch grundlegende logische Operationen erfolgt.

Als Ersatz für den fehlenden Augenblickswert werden viele Operationen auf den Mittelwert über N -Stichproben ausgelegt. Die Wahrscheinlichkeit von Einsen in diesen Stichproben ist $p_{1|2}$, wobei der Index $1|2$ entweder für 1 oder 2 steht. Für die logische Analyse ist die erwartete Anzahl von Einsen ($w_{1|2} = p_{1|2} \cdot N$) ein ganzzahliger Wert, was zu einem ganzzahligen Wert von Nullen führt ($z_{1|2} = (1 - p_{1|2}) \cdot N$). Mit steigender Stichprobengröße sinkt die Diskretisierungsschrittweite der Wahrscheinlichkeiten gegen Null.

Für die Operationen mit zwei Eingangsbitströmen wird eine statistische Verteilung der Positionen von Einsen in den Eingangsbitströmen angenommen. Dies führt zu dem Urnenproblem vom Typ „Ziehen ohne Zurücklegen“ zur Berechnung der Wahrscheinlichkeit von Einsen im Ausgabebitstrom [3].

OR Die logische Operation OR führt bei jeder Eins im ersten Eingangsbitstrom (BS_1) und jeder im zweiten Eingangsbitstrom (BS_2), die an der Position einer Null in BS_1 ist, zu einer Eins im Ausgabebitstrom (BS_{aus}). Die Wahrscheinlichkeit von Einsen im BS_{aus} ist somit mindestens die Wahrscheinlichkeit von Einsen in BS_1 , daraus folgt für die Einsen im Ausgabebitstrom bedingt durch den ersten Eingangsbitstrom

$$p_{OR_1} = \frac{w_1}{N} = p_1. \quad (3.12)$$

Um die Verteilung der Einsen im BS_{aus} zu berechnen, müssen die zusätzlichen Einsen von BS_2 , beschrieben werden. Aufgrund der bereits von BS_1 erzeugten Einsen, können nur die verbliebenen Nullen bedingt in Einsen umgewandelt werden.

Die Anzahl der kombinatorischen Möglichkeiten für eine Null im ersten Bitstrom ist gegeben durch

$$p_{OR_2} = \frac{(N - z_1)!}{N!} = \frac{(p_1 \cdot N)!}{N!}. \quad (3.13)$$

Die Möglichkeiten, g Einsen aus dem BS_2 an den Stellen von Nullen des BS_1 zu platzieren, berechnen sich wie folgt

$$p_{OR_3}(g) = \frac{z_1!}{t! \cdot (z_1 - g)!} = \frac{((1 - p_1) \cdot N)!}{g! \cdot ((1 - p_1) \cdot N - g)!}. \quad (3.14)$$

Dies schränkt die Anzahl der möglichen zusätzlichen Einsen ein. Der BS_{aus} kann weder mehr Einsen als Stellen haben, noch mehr als die Summe der Einsen in beiden Eingangsbitströmen. Dies führt zu einer maximalen Anzahl von zusätzlichen Einsen im Ausgangsbitstrom, die mathematisch wie folgt dargestellt werden kann

$$g_{OR,\max} = \min(z_1, w_2) = \min(1 - p_1, p_2) \cdot N. \quad (3.15)$$

Die Wahrscheinlichkeit, dass mehr Einsen im BS_{aus} sind, ist Null, da dieses Ereignis nicht möglich ist.

Die Wahrscheinlichkeit, dass g Einsen aus dem BS_2 mit Nullen des BS_1 kombiniert werden, ist gegeben durch

$$p_{OR_4}(g) = \frac{w_2!}{(w_2 - g)!} = \frac{(p_2 \cdot N)!}{(p_2 \cdot N - g)!}. \quad (3.16)$$

Die restlichen kopierten Nullen des BS_1 $((1 - p_1) \cdot N - g)$ müssen auch den Nullen des BS_2 entsprechen. Die Wahrscheinlichkeit für dieses Ereignis kann ausgedrückt werden durch

$$p_{OR_5}(g) = \frac{z_2!}{(z_2 - z_1 + t)!} = \frac{((1 - p_2) \cdot N)!}{((p_1 - p_2) \cdot N + g)!}. \quad (3.17)$$

BS_{aus} hat bei einer OR-Verknüpfung mindestens so viele Einsen, wie jeder einzelne Eingangsbitstrom. Dies kann ausgedrückt werden als

$$g_{OR,\min} = \max(0, w_2 - w_1) = \max(0, p_2 - p_1) \cdot N. \quad (3.18)$$

Die Einschränkungen durch Gleichung (3.15) und Gleichung (3.18) stellen sicher, dass die Fakultät nur aus den Werten $\in \mathbb{N}_0$ berechnet wird. Das Produkt aus Gleichung (3.13) bis Gleichung (3.17) in den Grenzen von Gleichung (3.15) und Gleichung (3.18) beschreibt die Wahrscheinlichkeit

von t zusätzlichen Einsen aus dem BS_2 im BS_{aus} , wenn die Eingangsbitströme durch die logische Operation OR verknüpft werden. Zusammen ergeben die Formeln

$$p_{OR}(g) = \frac{(p_1 \cdot N)!}{N!} \cdot \frac{((1 - p_1) \cdot N)!}{g! \cdot ((1 - p_1) \cdot N - g)!} \cdot \frac{(p_2 \cdot N)!}{(p_2 \cdot N - g)!} \cdot \frac{((1 - p_2) \cdot N)!}{((p_1 - p_2) \cdot N + g)!}. \quad (3.19)$$

Infolgedessen wird die Wahrscheinlichkeit einer Eins im BS_{aus} ($p_{y,OR}$) durch die Kombination von Gleichung (3.12) und Gleichung (3.19) gegeben

$$\begin{aligned} p_{y,OR} &= \sum_{g=\max(0, w_2 - w_1)}^{\min(z_1, w_2)} \left(\frac{g + w_1}{N} \right) \cdot p_{OR}(g) \\ &= \sum_{g=\max(0, p_2 - p_1) \cdot N}^{\min(1 - p_1, p_2) \cdot N} \frac{g}{N} \cdot p_{OR}(g) + p_1. \end{aligned} \quad (3.20)$$

Mit Gleichung (3.20) kann die Wahrscheinlichkeit einer Eins im BS_{aus} für Kombinationen von $p_{1|2} \in \{0, 1\}$ berechnet werden. In Abb. 3.23 sind die die erwarteten Mittelwerte für Einsen nach der logischen Operation OR abgebildet.

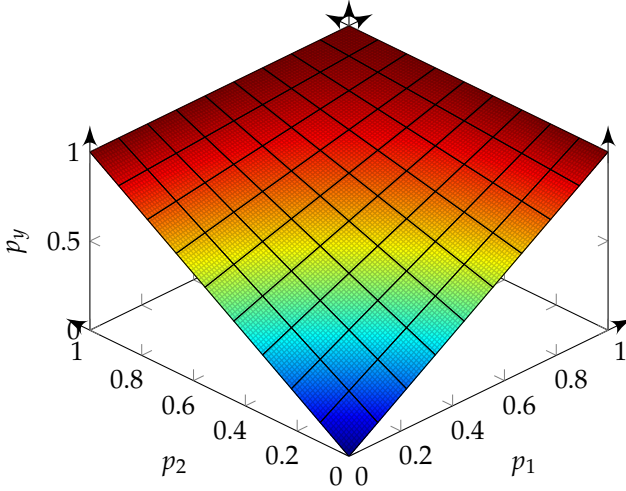


Abbildung 3.23: Erwartete Wahrscheinlichkeit einer Eins im Ausgangsbitstrom für die Eingangsbitströme BS_1 und BS_2 , mit der Wahrscheinlichkeit von Einsen $p_{1|2}$, mit OR verknüpft [Kle18].

AND Bei der Verknüpfung von BS_1 und BS_2 mit der Operation AND, müssen beide Bitströme eine Eins an der gleichen Position aufweisen, damit in BS_{aus} eine Eins ausgegeben wird. Somit ergibt sich die minimale Anzahl von Einsen im Ausgabebitstrom, wenn die Einsen in BS_1 mit Nullen in BS_2 kombiniert werden und umgekehrt. Die resultierenden, nicht mit Nullen kombinierbaren Einsen geben die minimale Anzahl der Einsen im Ausgabebitstrom an. Diese ergeben sich aus der Gesamtanzahl Einsen abzüglich der Gesamtlänge des betrachteten Ausschnitts. Als mathematischer Ausdruck lässt sich die minimale Anzahl von Einsen im BS_{aus} wie folgt beschreiben

$$g_{\text{AND,min}} = \max(0, w_1 + w_2 - N) = \max(0, p_1 + p_2 - 1) \cdot N. \quad (3.21)$$

Die maximale Anzahl von Einsen im BS_{aus} , ist die minimale Anzahl von Einsen in einem der Eingangsbitströme, da nur eine Eins ausgegeben wird,

wenn beide Eingangsbitströme an der Stelle eine Eins aufweisen. Dies kann ausgedrückt werden als

$$g_{AND,max} = \min(w_1, w_2) = \min(p_1, p_2) \cdot N. \quad (3.22)$$

Für die Beschreibung des Ereignisses, dass sich zwei Einsen von beiden Eingangsbitströmen an der gleichen Position befinden, werden die Einsen vom BS_1 betrachtet. Die Anzahl der kombinatorischen Möglichkeiten für eine Eins im BS_1 ist gegeben durch

$$p_{AND_1} = \frac{(N - w_1)!}{N!} = \frac{((1 - p_1) \cdot N)!}{N!}. \quad (3.23)$$

Die Möglichkeiten, g Einsen vom BS_2 an den gleichen Positionen, wie im BS_1 anzuordnen, sind gegeben wie folgt

$$p_{AND_2}(g) = \frac{w_1!}{g! \cdot (w_1 - g)!} = \frac{(p_1 \cdot N)!}{g! \cdot (p_1 \cdot N - g)!}. \quad (3.24)$$

Die Wahrscheinlichkeit, dass sich g Einsen aus dem BS_2 an den gleichen Positionen, wie diejenigen von BS_1 befinden, kann wie folgt beschrieben werden

$$p_{AND_3}(g) = \frac{w_2!}{(w_2 - g)!} = \frac{(p_2 \cdot N)!}{(p_2 \cdot N - g)!}. \quad (3.25)$$

Auf den Positionen der anderen Einsen im BS_1 sollten Nullen im BS_2 stehen. Die Wahrscheinlichkeit hierfür wird beschrieben durch

$$p_{AND_4}(g) = \frac{z_2!}{(z_2 - w_1 + g)!} = \frac{((1 - p_2) \cdot N)!}{((1 - p_1 - p_2) \cdot N + g)!}. \quad (3.26)$$

Das Produkt aus Gleichung (3.23), Gleichung (3.24), Gleichung (3.25) und Gleichung (3.26) in den Grenzen Gleichung (3.21) und Gleichung (3.22) beschreibt die Wahrscheinlichkeit von g Einsen im BS_{aus} , wenn die Eingangsbitströme mit der logischen Operation AND verknüpft werden

$$p_{AND}(g) = \frac{((1 - p_1) \cdot N)!}{N!} \cdot \frac{(p_1 \cdot N)!}{g! \cdot ((1 - p_1) \cdot N - g)!} \cdot \frac{(p_2 \cdot N)!}{(p_2 \cdot N - g)!} \cdot \frac{((1 - p_2) \cdot N)!}{((1 - p_1 - p_2) \cdot N + g)!}. \quad (3.27)$$

Die Wahrscheinlichkeit einer Eins im BS_{aus} ($p_{y,AND}$) ergibt sich aus Gleichung (3.27)

$$\begin{aligned}
 p_{y,AND} &= \sum_{g=\max(0,w_1+w_2-N)}^{\min(w_1,w_2)} \frac{g}{N} \cdot p_{AND}(g) \\
 &= \sum_{g=\max(0,p_1+p_2-1) \cdot N}^{\min(p_1,p_2) \cdot N} \frac{g}{N} \cdot p_{AND}(g).
 \end{aligned} \tag{3.28}$$

Mit Gleichung (3.28) kann die Wahrscheinlichkeit einer Eins im BS_{aus} für Kombinationen von $p_{1|2} \in \{0,1\}$ berechnet werden. Abb. 3.24 illustriert die Erwartungswerte der mit AND verknüpften Eingangsbitströme.

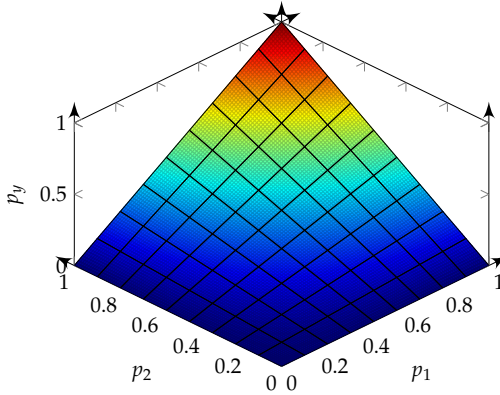


Abbildung 3.24: Erwartete Wahrscheinlichkeit einer Eins im Ausgangsbitstrom für die Eingangsbitströme BS_1 und BS_2 , mit der Wahrscheinlichkeit von $p_{1|2}$ für eine Eins, die mit AND verknüpft wurden [Kle18].

XOR Bei der Operation XOR ergibt sich eine Eins im BS_{aus} , wenn die Eingänge an dieser Position unterschiedliche logische Werte haben. Die logische Operation XOR zählt folglich die Unterschiede in den Bitströmen. Daher muss nur die Wahrscheinlichkeit einer Eins im BS_2 kombiniert mit einer Null im BS_1 beschrieben werden, um die Anzahl der Einsen im BS_{aus} zu berechnen, denn daraus folgt die Anzahl der Einsen in BS_1 , die mit einer Null in BS_2 kombiniert werden.

Die minimale Anzahl dieses Ereignisses ist somit die Differenz zwischen den Einsen im BS_2 und denen im BS_1 . Der Unterschied lässt sich wie folgt beschreiben

$$g_{XOR,min} = \max(0, w_2 - w_1) = \max(0, p_2 - p_1) \cdot N. \quad (3.29)$$

Die maximale Anzahl von kombinierten Einsen des BS_2 mit Nullen des BS_1 wird durch deren Anzahl begrenzt. Somit ergibt sich die maximale Anzahl von möglichen Kombinationen zu

$$g_{XOR,max} = \min(z_1, w_2) = \min(1 - p_1, p_2) \cdot N. \quad (3.30)$$

Definition 1. Die Funktion $parity(x)$ von $\mathbb{N}_0 \rightarrow \mathbb{N}_0$: $parity(x) = x \bmod 2$.

Weiterhin muss eine Besonderheit der Operation XOR berücksichtigt werden. Es ist nicht möglich die Parität von Einsen zu ändern ($parity((p_1 + p_2) \cdot N) = parity(p_y \cdot N)$), wobei $parity(x)$ in ?? 1 definiert ist. Das statistische Modell berücksichtigt diese Eigenschaft nicht und gibt für diese Ereignisse die gleiche Wahrscheinlichkeit an, wie für die Ereignisse mit der richtigen Parität. Daraus folgt, dass die Wahrscheinlichkeiten der Ereignisse mit unterschiedlichen Paritäten auf Null gesetzt werden müssen. Dies wird erreicht durch Multiplikation mit

$$(1 - |parity(g) - parity((p_1 + p_2) \cdot N)|) \quad (3.31)$$

Zuerst wird die Anzahl der Möglichkeiten beschrieben, um die Nullen im BS_1 zu platzieren

$$p_{XOR_1} = \frac{(N - z_1)!}{N!} = \frac{(p_1 \cdot N)!}{N!}. \quad (3.32)$$

Die Möglichkeiten, g -Einheiten aus dem BS_2 an den Stellen von Nullen aus dem BS_1 anzuordnen, können wie folgt beschrieben werden

$$p_{XOR_2}(g) = \frac{z_1!}{g! \cdot (z_1 - g)!} = \frac{((1 - p_1) \cdot N)!}{g! \cdot ((1 - p_1) \cdot N - g)!}. \quad (3.33)$$

Die Wahrscheinlichkeit, dass g Einsen aus dem BS_2 auf Stellen von Nullen im BS_1 fallen, ist gegeben durch

$$p_{XOR_3}(g) = \frac{w_2!}{(w_2 - g)!} = \frac{(p_2 \cdot N)!}{(p_2 \cdot N - g)!}. \quad (3.34)$$

Die restlichen Nullen des BS_1 $((1 - p_1) \cdot N - g)$ sollten ebenfalls an den selben Positionen wie Nullen im BS_2 sein. Die Wahrscheinlichkeit für dieses Ereignis kann ausgedrückt werden durch

$$p_{XOR_4}(g) = \frac{z_2!}{(z_2 - z_1 + g)!} = \frac{((1 - p_2) \cdot N)!}{((p_1 - p_2) \cdot N + g)!}. \quad (3.35)$$

Das Produkt aus Gleichung (3.32) bis Gleichung (3.35) mit der Paritätsprüfung Gleichung (3.31) in den Grenzen von Gleichung (3.29) und Gleichung (3.30) führt zu der Wahrscheinlichkeit für g solcher Ereignisse. Mathematisch folgt somit

$$\begin{aligned} p_{XOR}(g) = & \frac{(p_1 \cdot N)!}{N!} \cdot \frac{((1 - p_1) \cdot N)!}{g! \cdot ((1 - p_1) \cdot N - g)!} \cdot \\ & \cdot \frac{(p_2 \cdot N)!}{(p_2 \cdot N - g)!} \cdot \frac{((1 - p_2) \cdot N)!}{((p_1 - p_2) \cdot N + g)!} \cdot \\ & \cdot (1 - |\text{parity}(g) - \text{parity}((p_1 + p_2) \cdot N)|). \end{aligned} \quad (3.36)$$

Die Wahrscheinlichkeit einer Eins im BS_{aus} ($p_{y,XOR}$) kann berechnet werden. Die Anzahl der Einsen im BS_2 kombiniert mit Nullen aus dem BS_1 wird mit g bezeichnet. Daraus folgt, dass BS_{aus} g Einsen enthalten muss. Die restlichen Nullen des BS_1 werden mit Nullen des BS_2 kombiniert. Diese fügen dem BS_{aus} keine weiteren Einsen hinzu. Im Gegensatz dazu fügen die überschüssigen Einsen im BS_1 , die an der gleichen Position, wie Nullen im BS_2 sind, weitere Einsen im BS_{aus} hinzu. Dieses Ereignis tritt für alle Einsen im BS_1 (w_1) abzüglich der übrigen Einsen im BS_2 ($w_2 - g$) auf, da g Einsen des BS_2 bereits mit Nullen aus dem BS_1 kombiniert sind. Dies kann beschrieben werden als

$$n_{\text{ones},y} = g + w_1 - w_2 + g = 2 \cdot g + (p_1 - p_2) \cdot N \quad (3.37)$$

Die Wahrscheinlichkeit einer Eins im BS_{aus} kann mit Gleichung (3.36) und Gleichung (3.37) berechnet werden

$$\begin{aligned}
 p_{y,XOR} &= \sum_{g=\max(0,w_2-w_1)}^{\min(z_1,w_2)} \left(\frac{2 \cdot g + w_1 - w_2}{N} \right) \cdot p_{XOR}(g) \\
 &= \sum_{g=\max(0,p_2-p_1) \cdot N}^{\min(1-p_1,p_2) \cdot N} \frac{2 \cdot g}{N} \cdot p_{XOR}(g) + p_1 - p_2.
 \end{aligned} \tag{3.38}$$

Mit Gleichung (3.38) kann die Wahrscheinlichkeit einer Eins im BS_{aus} für Kombinationen von $p_{1|2} \in \{0,1\}$ berechnet werden. Abb. 3.25 illustriert die Wahrscheinlichkeit einer Eins im BS_{aus} nach der Verknüpfung der Eingangsbitströme mit der logischen Operation XOR.

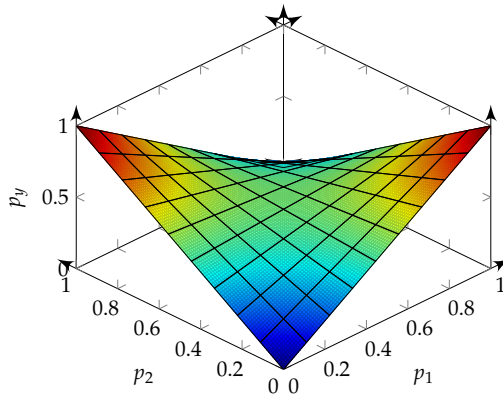


Abbildung 3.25: Erwartete Wahrscheinlichkeit einer Eins im Ausgangsbitstrom für die XOR verknüpften Eingangsbitströme BS_1 und BS_2 , mit der Wahrscheinlichkeit von $p_{1|2}$ für eine Eins [Kle18].

Mit den Gleichungen Gleichung (3.20), Gleichung (3.28) und Gleichung (3.38) kann die Wahrscheinlichkeit von Einsen am Ausgang einer logischen Operation für statistisch unabhängige Eingangsbitströme abgeleitet werden. Im nächsten Abschnitt werden mathematische Funktionen für die Abbildung der Wahrscheinlichkeiten von Einsen in den Eingangsbit-

strömen auf Wahrscheinlichkeiten von Einsen in den Ausgangsbitströmen vorgestellt.

3.2.2.3.2 Mathematische Funktionen Um die grundlegenden logischen Operationen als mathematische Operationen für $\Delta\Sigma$ SV nutzen zu können, muss eine mathematische Funktion entwickelt werden, die die gleiche Zuordnung der Eingangswerte zu den Ausgangswerten erzeugt.

Die in Abschnitt 3.2.2.3.1 entwickelte statistische Beschreibung führt zu erwarteten Wahrscheinlichkeiten von Einsen in den BS_{aus} der logischen Operationen OR, AND und XOR für die BS_1 und BS_2 mit der Wahrscheinlichkeit $p_{1|2} \in \{0 \dots 1\}$ für Einsen. Wie in Abschnitt 3.2.2.3.1 dargestellt, kann der von einem Bitstrom repräsentierte Wert durch Mittelwertbildung interpretiert werden. Die erwartete Anzahl an Einsen in N abgetasteten Bits eines Bitstroms stellt die gleiche Information dar, wie die durchschnittliche Wahrscheinlichkeit. Somit steht der erwartete Prozentsatz von Einsen im Bitstrom in direktem Zusammenhang mit dem dargestellten Wert.

Mit der numerischen Interpretation der Eingangs- und Ausgangsbitströme, wie in Kapitel 2 dargestellt, können mathematische Funktionen zu den Transformationen der logischen Operationen entwickelt werden.

OR In Abb. 3.23 auf Seite 34 ist die Wahrscheinlichkeit für Einsen im BS_{aus} der logischen Operation OR über den Wahrscheinlichkeiten von Einsen in BS_1 und BS_2 dargestellt. Bei ausschließlich Nullen in BS_1 oder BS_2 zeigt die Wahrscheinlichkeit von Einsen einen linearen Zusammenhang zu der Wahrscheinlichkeit von Einsen in dem anderen Eingangsbitstrom. Dieses Verhalten lässt sich, bei unipolarer Interpretation der Bitströme, mit der Funktion $BS_1 + BS_2$ beschreiben. Für den Fall, dass BS_1 oder BS_2 nur Einsen enthält, führt die Funktion für jede Eins im anderen Bitstrom zu einem Überlauf. Die Ausgangswahrscheinlichkeit der Logikfunktion ist jedoch begrenzt. Somit muss der Anteil überschüssiger Einsen entfernt werden. Für den Randbereich, bei dem ein Eingang nur Einsen enthält, lässt sich dies mit der Subtraktion von $BS_1 \cdot BS_2$ erreichen, da der Bitstrom, der nur Einsen enthält dem Wert Eins zugeordnet ist. Diese zusammengefasste unipolare Funktion $BS_1 + BS_2 - BS_1 \cdot BS_2$ erzeugt die gleiche Abbildung,

wie die erwarteten Wahrscheinlichkeiten für Einsen, wie aus Abb. 3.26 und Abb. 3.23 auf Seite 34 ersichtlich ist.

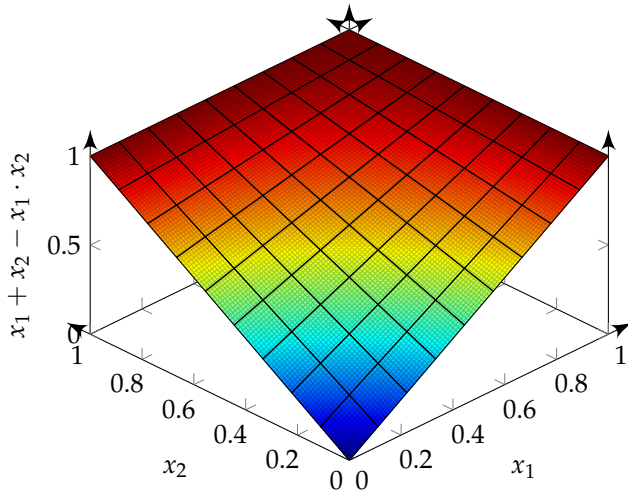


Abbildung 3.26: Abbild der Funktion $BS_1 + BS_2 - BS_1 \cdot BS_2$ [Kle18].

AND In Abb. 3.24 auf Seite 36 wird die Wahrscheinlichkeit von Einsen im BS_{aus} der logischen Operation AND für verschiedene Wahrscheinlichkeiten von Einsen in BS_1 und BS_2 dargestellt. Liegen an einem Eingang nur Nullen an, enthält BS_{aus} nur Nullen. Bei dem Fall, dass an einem Eingang nur Einsen anliegen, weist der Ausgang eine lineare Abhängigkeit von der Wahrscheinlichkeit der Einsen des anderen Eingangs auf. Dies führt zur unipolaren mathematischen Funktion $BS_1 \cdot BS_2$. Die erwartete Wahrscheinlichkeit von Einsen im Vergleich zu Abb. 3.27 zeigt, dass sie das gleiche Bild ergeben.

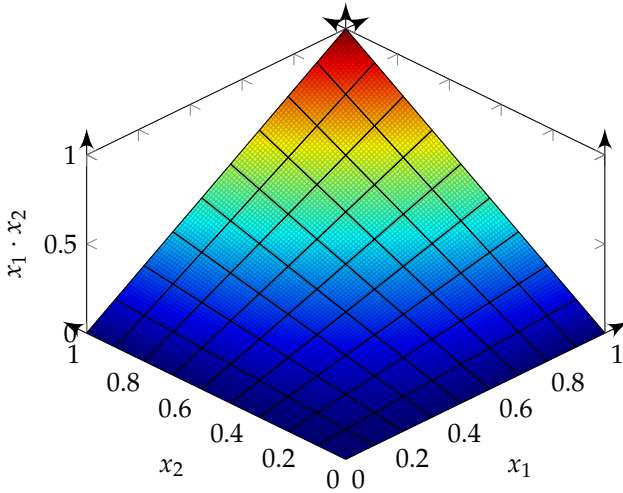


Abbildung 3.27: Abbild der Funktion $BS_1 \cdot BS_2$ [Kle18].

XOR In Abb. 3.25 auf Seite 39 wird der Prozentsatz von Einsen im BS_{aus} der logischen Operation XOR für verschiedene Wahrscheinlichkeiten von Einsen in BS_1 und BS_2 gezeigt. In diesem Fall ergibt sich ein Sattelpunkt bei $p_{\text{aus}} = 0,5$ für p_1 oder p_2 gleich $0,5$. Dieses Verhalten entspricht der Interpretation, dass $p = 0,5$ gleich Null ist und die bipolare Funktion $BS_1 \cdot BS_2$ ist. Zusätzlich zeigt sich eine lineare Abhängigkeit von der Wahrscheinlichkeit von Einsen an einem Eingang, wenn der andere Eingang konstant Nullen oder Einsen aufweist. Die Maxima treten auf, wenn die Bitströme unterschiedlich sind und die Minima, wenn sie gleich sind. Diese Extrempunkte werden bei einer Multiplikation erwartet, allerdings sind die Maxima und die Minima vertauscht. Das führt zur bipolaren Funktion $-BS_1 \cdot BS_2$. Die Abbildung der bipolaren Funktion ist gleich der Abbildung der Wahrscheinlichkeit von Einsen im BS_{aus} der logischen Operation XOR, wie in Abb. 3.28 und Abb. 3.25 auf Seite 39 ersichtlich ist.

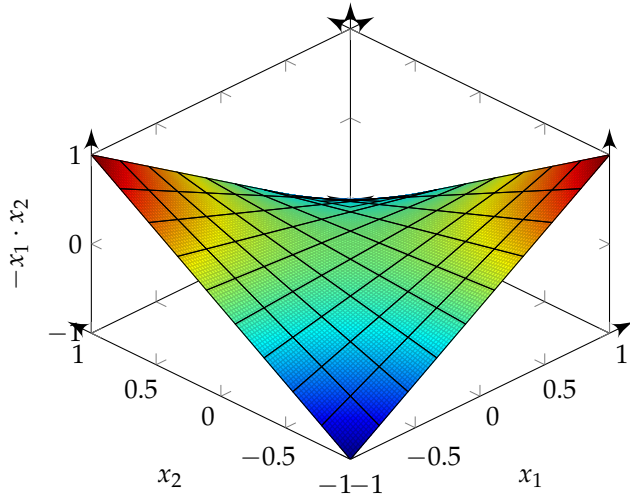


Abbildung 3.28: Abbild der Funktion $-BS_1 \cdot BS_2$ [Kle18].

Anhand der erwarteten Mittelwerte wurden folgende mathematischen Abbildungen entwickelt

$$f_{\text{or}}(BS_1, BS_2) = BS_1 + BS_2 - BS_1 \cdot BS_2 : \forall BS_1, BS_2 \in [0 \dots 1] \quad (3.39)$$

$$f_{\text{and}}(BS_1, BS_2) = BS_1 \cdot BS_2 : \forall BS_1, BS_2 \in [0 \dots 1] \quad (3.40)$$

$$f_{\text{xor}}(BS_1, BS_2) = -BS_1 \cdot BS_2 : \forall BS_1, BS_2 \in [-1 \dots 1] \quad (3.41)$$

3.2.2.3.3 Standardabweichung Diese Funktionen sollen für die $\Delta\Sigma SV$ verwendet werden. Die allgemeine Funktion wird durch den Mittelwert definiert. Um die Abbildungsqualität zu beurteilen ist die Standardabweichung ein geeignetes Maß. Daher wird die Standardabweichung detailliert betrachtet.

Die Standardabweichung ist in der Statistik neben dem Mittelwert eine wichtige Kennzahl, in diesem Fall kann sie auch als Maß für die Bildqualität der abgeleiteten mathematischen Funktion verwendet werden. Nach

den Gesetzen der Logik wird die Standardabweichung nach [32] berechnet

$$\sigma = \sqrt{\sum_{k=0}^n (p(k) \cdot k - p_y)^2}. \quad (3.42)$$

Basierend auf Gleichung (3.42) kann die Verteilung der Ergebnisse der logischen Operationen berechnet werden. Bei der Verknüpfung OR und AND sind die Standardabweichungen gleich. Der Verlauf der Standardabweichung ist in Abb. 3.29 abgebildet.

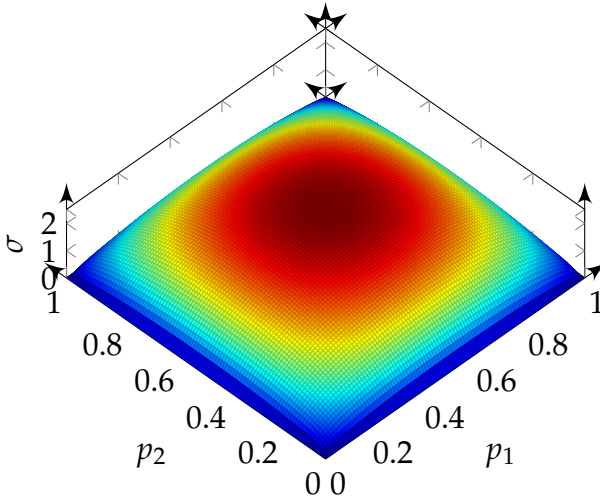


Abbildung 3.29: Standardabweichung für die Wahrscheinlichkeit einer Eins im BS_{aus} für die BS_1 und BS_2 , mit der Wahrscheinlichkeit von Einsen $p_{1|2}$, verknüpft mit OR oder AND [Kle18].

Bei der Verknüpfung XOR wird die Standardabweichung gegenüber den Operationen durch OR und AND verdoppelt. Die Standardabweichung der Operation XOR ist in Abb. 3.30 abgebildet.

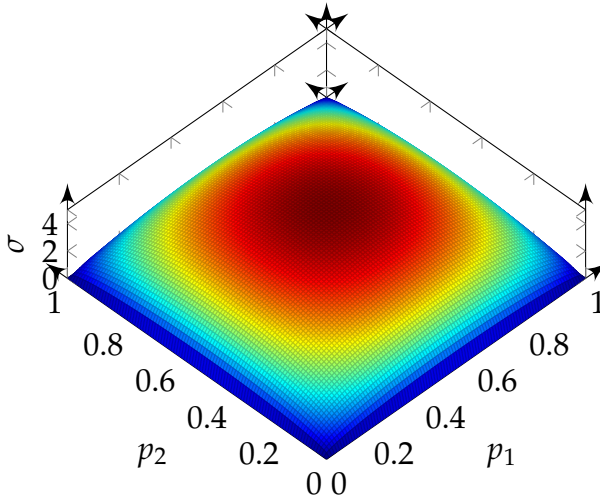


Abbildung 3.30: Standardabweichung für die Wahrscheinlichkeit einer Eins im BS_{aus} für die BS_1 und BS_2 , mit der Wahrscheinlichkeit von Einsen $p_{1|2}$, verknüpft mit XOR [Kle18].

Die Form der Standardabweichung ist für alle drei logischen Operationen gleich. An den Rändern der dargestellten Ebene gibt es keine kombinatorischen Möglichkeiten, da mindestens ein Bitstrom (BS_1, BS_2) nur aus Nullen oder Einsen besteht. Somit folgt, dass das Ergebnis nur von der Wahrscheinlichkeit einer Eins des anderen Bitstroms abhängig ist. In der Mitte der Fläche, wo beide Bitströme jeweils zur Hälfte aus Einsen und Nullen bestehen, gibt es erwartungsgemäß die meisten kombinatorischen Möglichkeiten. In dem einen Extremfall ist jede Eins des einen Bitstroms an der gleichen Position, wie eine Eins des anderen Bitstroms, so ergeben sich für die logischen Operationen OR und AND Ausgangsbitströme, die den Eingangsbitströmen entsprechen. Für die logische Operation XOR ergibt sich ein Bitstrom aus Nullen als Ausgangsbitstrom. In dem anderen Extremfall, wenn jede Eins aus dem einen Bitstrom an der gleichen Position ist, wie auf eine Null des anderen Bitstroms, ergibt sich für die logische Operation AND ein Ausgangsbitstrom aus Nullen. Für die logischen Operationen OR und XOR ergibt sich ein Ausgangsbitstrom, der aus Einsen besteht. Somit sind an diesem Punkt die größten Standardabweichungen

zu erwarten. Aufgrund der Tatsache, dass der Ausgangsbitstrom bei XOR von Null zu Eins wechselt und bei OR und AND nur von 0,5 zu Null oder Eins, ist die Skalierung um den Faktor Zwei der Standardabweichungen von OR und AND zu XOR zu erwarten. Dies führt zu der Erwartung, dass die Abbildungsqualität an den Definitionsgrenzen der Eingangsgrößen sehr gut ist und sich zur Mitte verschlechtert.

Zusammenfassend wird der Einfluss der Verknüpfungen OR, AND und XOR auf den Mittelwert zweier Eingangsbitströme mit unabhängigen Hochfrequenzanteilen auf den Ausgangsbitstrom und die Standardabweichung dieser Operationen diskutiert.

3.2.2.3.4 Quantisierungsrauschen In diesem Abschnitt wird die Unabhängigkeit des Quantisierungsrauschens von zwei Bitströmen diskutiert, das zu den hochfrequenten Signalkomponenten führt.

Der Mittelwert und die Standardabweichung nach logischen Operationen sind für statistisch verteilte Bitströme ermittelt. Diese Annahme basiert auf dem linearen Modell des $\Delta\Sigma$ -M, wie in Abb. 3.31 dargestellt [2].

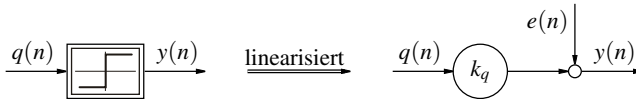


Abbildung 3.31: Lineares Modell des Quantisierers eines $\Delta\Sigma$ -M, bestehend aus einer Verstärkung (k_q) und einer weißen Rauschquelle ($e(n)$) [28].

In der Literatur wird zur Analyse eines $\Delta\Sigma$ -Ms mit Ordnungen größer 1 das linearisierte Modell [2, 28, 31, 34] [HKKS16] verwendet. Bei nichtlinearen Operationen in $\Delta\Sigma$ SV ist die Annahme einer weißen Rauschquelle ungeeignet, da die einzelnen Bits bei jedem Taktzyklus einen Einfluss auf den Ausgangsbitstrom haben, während die Annahme einer weißen Rauschquelle für eine größere Zeitskala gültig ist. Daher müssen die Rauschquellen der verwendeten $\Delta\Sigma$ -M unabhängig sein, oder die berechneten Wahrscheinlichkeiten und Standardabweichungen entsprechen nicht den real auftretenden.

Das hochfrequente Rauschen verschiebt die Position einzelner Einsen und Nullen lokal im Bitstrom, allerdings hat es keinen Einfluss auf die im Nutzfrequenzband übertragene Information. Um die statistische Unabhängigkeit des Rauschens zweier Bitströme abzuschätzen, kann die normierte Kreuzkorrelation ohne Verschiebung Gleichung (3.43) mit der vorzeichenbehafteten Interpretation der Bitströme verwendet werden.

$$K_{BS_1, BS_2} = \frac{1}{n} \sum_{i=0}^n x_{1_{\{-1;1\}}}(i) \cdot x_{2_{\{-1;1\}}}(i) \quad (3.43)$$

Die Bewertung der vorgestellten Ansätze zur Variation der hochfrequenten Signalanteile wird mit einer sinusförmigen Anregung untersucht, wie in Abb. 3.32 gezeigt.

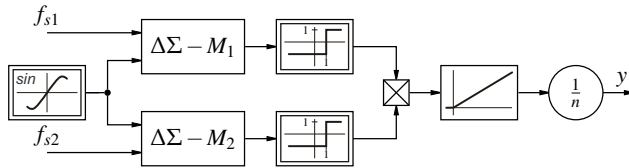


Abbildung 3.32: Simulation zur Untersuchung der Verkopplung des Rauschens von $\Delta\Sigma$ -M [Kle18].

Die Kreuzkorrelation gibt die Verknüpfung zweier Signale an. Daraus folgt, dass ein möglichst geringer absoluter Wert von K_{BS_1, BS_2} für die statistische Unabhängigkeit steht. Die stärkste Korrelation weisen zwei digitale $\Delta\Sigma$ -M mit gleicher Taktfrequenz, gleichen Parametern und gleichem Eingangssignal auf. In diesem Fall beträgt $K_{BS_1, BS_2} = 1$, was erwartet wird, da das Rauschen durch den Quantisierer erzeugt wird.

Aufgrund der Nichtlinearität des $\Delta\Sigma$ -M, können kleine Unterschiede in den Eingangssignalen zu einem geringeren K_{BS_1, BS_2} führen. Der Nachteil dieser Lösung ist, dass zum Einen die Unabhängigkeit des Rauschens für unterschiedliche Signale nicht garantiert ist und zum Anderen die verarbeiteten Signale nicht zwangsläufig unterschiedlich sind. Durch eine Reduktion des möglichen Signalbereichs und einer zusätzlichen Abbildungen lässt sich garantieren, dass die Signale unterschiedlich sind.

Ein anderer Ansatz ist, unterschiedliche Parameter bei den $\Delta\Sigma$ -M zu

verwenden [HKS16], zum Beispiel [34] und [25]. Mit dieser Kombination wird im Versuch ein $K_{BS_1,BS_2} = 0,278$ erreicht.

Eine weitere Möglichkeit das lokale Rauschen zu beeinflussen, ist unterschiedliche Abtastfrequenzen zu verwenden ($f_{s1} \neq f_{s2}$). Beispielhaft wird ein $\Delta\Sigma$ -M mit der doppelten Abtastfrequenz des anderen betrieben, wobei beide nach [25] ausgelegt sind. Diese Implementierung resultiert in $K_{BS_1,BS_2} = 0,298$. Dabei hat sie den Nachteil, dass Teile des Systems mit höheren Taktfrequenzen betrieben werden müssen, als für die resultierende Bandbreite nötig ist.

Die Implementierung mit unterschiedlich ausgelegten $\Delta\Sigma$ -M erzielt die höchste Unabhängigkeit der hochfrequenten Rauschsignalkomponenten. Mit anderen Auslegungen kann wahrscheinlich ein geringeres K_{BS_1,BS_2} erzielt werden. Daher ist die vorgeschlagene Lösung, unterschiedlich gestaltete $\Delta\Sigma$ -M zu verwenden.

3.2.2.3.5 Ergebnisse Der Versuchsaufbau in Abb. 3.33 moduliert die Eingangssignale mit zwei unterschiedlich gestalteten $\Delta\Sigma$ -M. Die resultierenden Bitströme werden mit einer logischen Operation mit zwei Eingängen (AND, OR oder XOR) verknüpft und mit einem sinc^3 -Filter [34] ausgewertet. Der resultierende Multibit-Wert wird mit dem sinc^3 gefilterten Multibit-Wert der entsprechenden Multibit-Operation verglichen. Für die unipolaren Funktionen AND und OR werden unipolare Signalquellen und $\Delta\Sigma$ -M verwendet und die bipolare Funktion XOR wird durch bipolare Eingangssignale mit bipolaren $\Delta\Sigma$ -M angesteuert.

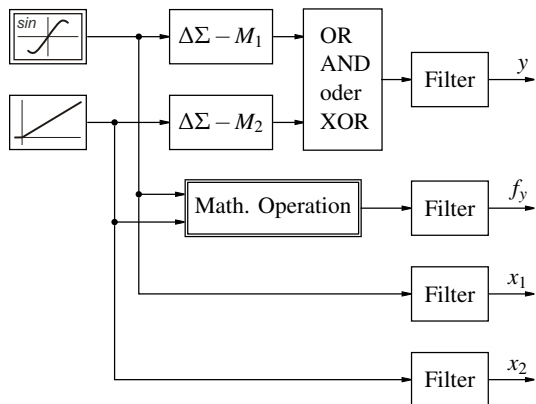


Abbildung 3.33: Versuchsaufbau zur Bewertung der aus logischen Operationen resultierenden Funktionen [Kle18].

Die Simulationsparameter sind in Tabelle 3.10 abgedruckt.

Tabelle 3.10: Simulationsparameter [KS18].

Parameter	Wert
Abtastfrequenz	10 MHz
Simulationsdauer	1 s
Rüstzeit	50 μ s
Sinus Amplitude	1
Sinus Frequenz	10 Hz
Rampenstartwert	-1
Rampensteigung	2 s ⁻¹
Auswertungs sinc ³ -Filterlänge	128
Dezimierungsrate	1250

Die gefilterten Signalquellen sind in Abb. 3.34 dargestellt. Die Abbildungen der übertragenen Informationen durch OR und AND werden mit der Simulation, siehe Abb. 3.34, nachgewiesen. Dennoch treten die größten Abweichungen nicht in der Region um 0,5 s auf, wo sie aus den Überlegungen in Abschnitt 3.2.2.3.3 erwartet werden. Der resultierende Wert für die logische Operation OR weicht am meisten um 0,8 s und bei der logischen Funktion AND im gespiegelten Bereich um 0,2 s ab. Diese Tatsache zeigt, dass die Annahme statistisch entkoppelter Bitströme nicht für alle

Arbeitspunkte gültig ist. Daraus folgt die Notwendigkeit, die Abweichung der Operation für verschiedene $\Delta\Sigma$ -M Implementierungen auf einem Prüfstand mit den erwarteten Eingangssignalen zu bewerten.

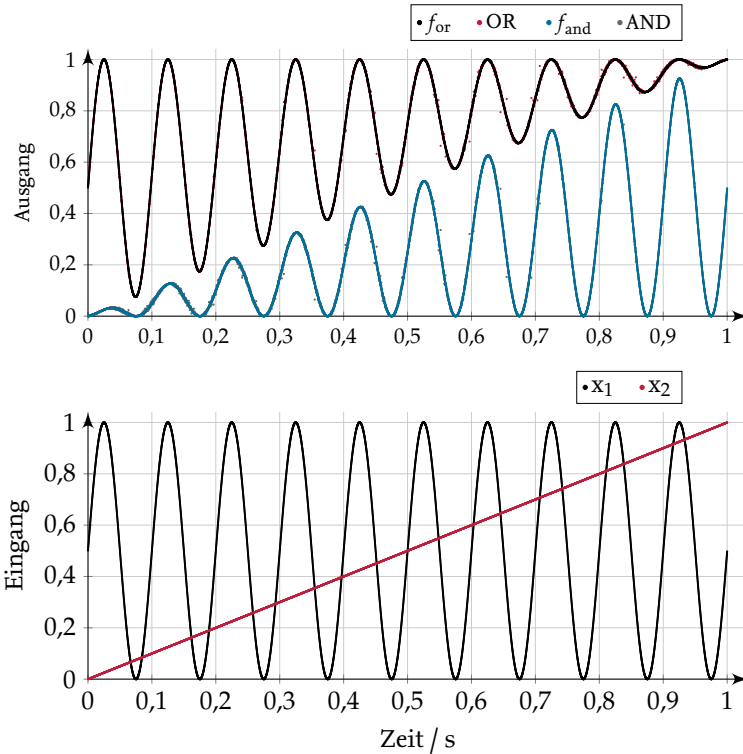


Abbildung 3.34: Abbildungen durch die Logikoperationen OR und AND und die entsprechenden Operationen $f_{or} = BS_1 + BS_2 - BS_1 \cdot BS_2$ und $f_{and} = BS_1 \cdot BS_2$ [Kle18].

Das Ergebnis in Abb. 3.35 für die Logikfunktion XOR zeigt das erwartete Verhalten. Bei Eingangssignalen an der Definitionsgrenze, zum Beispiel Beginn und Ende der Simulation, folgt der dargestellte Wert im Ausgangsbitstrom dem erwarteten Signal. Die Abweichungen sind am größten, wenn beide Eingangssignale einen Wert in der Mitte des Definitionsbereichs darstellen, der im Bereich um 0,5s auftritt.

Die Simulationsergebnisse zeigen, dass die statistischen Überlegungen gültig sind. Die daraus resultierenden Abweichungen sind geringer als erwartet. Um die logischen Verknüpfungen als mathematische Funktionen nutzen zu können, sollte die Implementierung der $\Delta\Sigma$ -M anhand der erwarteten Eingangssignale verifiziert werden.

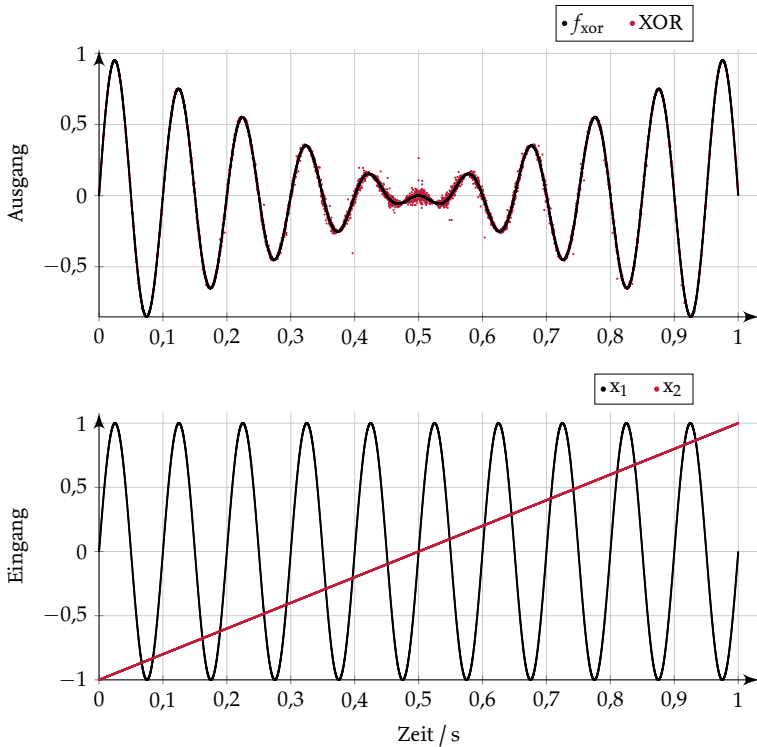


Abbildung 3.35: Abbildungen der Logikoperation XOR und die entsprechende mathematische Funktion $f_{xor} = -BS_1 \cdot BS_2$ [Kle18].

3.2.2.4 Evaluation

Da nichtlineare Funktionen das Frequenzspektrum verändern, ist die SINAD Berechnung komplizierter als bei linearen Operationen. Daher liegt der Fokus bei der Bewertung nichtlinearer Funktionen auf der Abbildungsqualität und der Gruppenlaufzeit. Die Abbildungsqualität

wird über die Varianz und die Gruppenlaufzeit mit der Sprungantwort gemessen, analog zu den linearen Operationen. In dieser Arbeit werden nur Funktionen mit einem oder zwei Eingängen behandelt, auch wenn einige auf weitere Eingänge erweitert werden können. Für die Messung der Abbildungsqualität wird der erste Eingang mit einem sinusförmigen Signal angeregt und wenn die Operation einen zweiten Eingang hat, wird dieser mit einem rampenförmigen Signal angeregt, wie in Abb. 3.36 dargestellt. Bei einer Funktion mit zwei Eingängen wird die Sprungantwort unabhängig für jeden Eingang mit einem konstanten Wert am anderen Eingang gemessen.

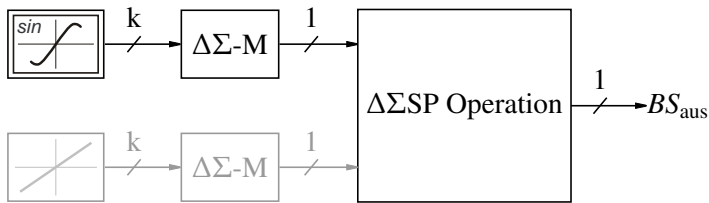


Abbildung 3.36: Testaufbau für nichtlineare Funktionen mit optionalem zweiten Eingang [KS18].

Alle Implementierungen sind in Festkommaarithmetik berechnet, da sich die Ergebnisse aus der Simulation auf einem Computer und dem Betrieb auf einem Field Programmable Gate Array (FPGA) nicht unterscheiden. Dies liegt daran, dass beide rein digital berechnet werden, weshalb die Simulationsergebnisse diskutiert werden. Die Parameter sind in Tabelle 3.11 abgedruckt. Die Rüstzeit beschreibt zu Beginn eine zusätzliche Simulationszeit, die nicht ausgewertet wird. Der rampenförmige Eingang ist für die Rüstzeit konstant und die sinusförmige Quelle wird phasenverschoben, um einem Phasenwinkel von 0° zu Beginn der Aufnahme zu erreichen. Die Filterlänge sinc^3 beschreibt die Anzahl der in jeder Stufe verarbeiteten Abtastwerte.

Tabelle 3.11: Simulationsparameter [KS18].

Parameter	Wert
Abtastfrequenz	10 MHz
Simulationsdauer	0,10 s
Rüstzeit	50 μ s
Sinus Amplitude	0,75
Sinus Frequenz	100 Hz
Rampenstartwert	-0,75
Rampensteigung	15 s ⁻¹
DSV sinc ³ -Filterlänge	4
Auswertungs sinc ³ -Filterlänge	128

3.2.2.5 Ergebnisse

Die Signalverarbeitung läuft mit einer Abtastfrequenz von 10 MHz. Abb. 3.37 zeigt die sinc³ gefilterten Ergebnisse für die verschiedenen Implementierungen der Multiplikation. Wie man sieht, erzeugt die modifizierte Delta Adder-basierte Implementierung hauptsächlich Rauschen, aber nicht das erwartete Signal. Das resultierende Signal korreliert nur bei Maxima und Minima mit dem erwarteten Signal, wobei in beiden Fällen eine positive Auslenkung entsteht. Die Signalqualität nimmt stark ab, da die modifizierte Struktur einen inhärenten Nachteil hat, nämlich die Kopplung der Eingangsbitströme, was auch für invertierte XOR logikbasierte Operationen ein Problem darstellt, wie bereits in [Kle18] erwähnt. Unter Verwendung eines zweiten Schieberegisters für den zweiten Eingangsbitstrom werden die gleichen Bit-Tupel durch die Operation geschoben. Diese Kopplung führt zunächst zur Multiplikation und anschließend zur Filterung. Dadurch wird die Rauschverformung, wie in Abb. 3.20 auf Seite 28 gezeigt, aufgehoben. Daher entfällt die Implementierung mit zwei Bit-Shift-Registern in der weiteren Betrachtung.

Die Parameter für die Simulation der Multiplikation sind in Tabelle 3.12 auf Seite 55 und der Ressourcenverbrauch auf einem FPGA ist in Tabelle 3.13 auf Seite 55 dargestellt. Die Implementierung basierend auf logischen Verknüpfungen folgt dem gewünschten Ausgangssignal mit etwas

Rauschen. Die Varianz ist drei Größenordnungen größer als bei der Implementierung auf Basis von Delta Addierern oder DSV. Der große Vorteil ist die kurze Gruppenlaufzeit. Mit sechs Taktzyklen ist sie doppelt so schnell wie die auf DSV basierende Implementierung. Die Implementierung auf Basis von Delta Addierern kommt dieser Gruppenlaufzeit nahe, wenn man den Durchschnitt betrachtet. Wie in Tabelle 3.14 auf Seite 57 angegeben, hat der erste Eingang eine Gruppenlaufzeit von 17 Taktzyklen und der zweite Eingang weist keine Gruppenlaufzeit auf. Die auf DSV basierende Implementierung führt zu der besten Varianz. Die optimale Implementierung für die Operationsmultiplikation hängt somit von den Anforderungen ab. Wenn die höchste Priorität auf dem Logikkatterverbrauch liegt und die Signalqualität nicht wichtig ist, ist die Implementierung auf Basis von Logikoperationen zu bevorzugen. Wenn zwei Signale multipliziert werden, von denen sich mindestens eines langsam ändert, kann die Implementierung auf Basis von Delta Addierern vorteilhaft sein, insbesondere wenn keine DSV Blöcke der FPGA verfügbar sind. Wenn die Signalqualität die höchste Priorität hat oder die Gruppenlaufzeit für beide Signale gleich sein sollte, ist die Implementierung auf Basis von DSV am besten geeignet.

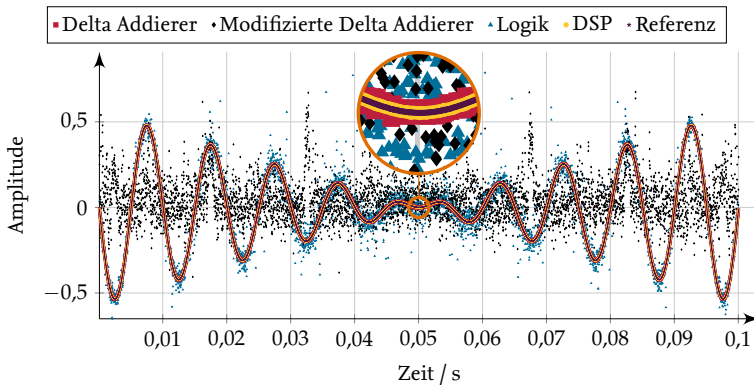


Abbildung 3.37: Tiefpass gefilterte Ergebnisse der Operation Multiplikation mit einer Dezimierungsrate von 200 [KS18].

Tabelle 3.12: Varianz und Gruppenlaufzeit der Operation Multiplikation (Zweiter Eingang, falls unterschiedlich) [KS18].

Implementierung	Varianz	Gruppenlaufzeit
Delta Addierer	5,2934e-6	17 (0)
Logik Operation	3,4792e-3	6
DSV	1,8948e-7	12

Tabelle 3.13: Ressourcenverbrauch der Operation Multiplikation auf einem FPGA [KS18].

Implementierung	Logikgatter	Multiplizierer (9x9)
Delta Addierer	318	0
Logik Operation	2	0
DSV	243	1

3.2.3 Begrenzung

Eine weitere wichtige nichtlineare algebraische Operation in Regelsystemen ist die Begrenzung. Um den Wert eines Bitstroms innerhalb des gültigen Darstellungsbereichs weiter zu begrenzen, muss der angenäherte Momentanwert durch Mittelung ausgewertet werden. Wenn der gefilterte Wert gültig ist, hat der Ausgang bereits die vorgesehene Begrenzung überschritten und hätte in vorhergehenden Taktschritten begrenzt werden müssen.

3.2.3.1 Schieberegister

Das Problem der Mittelwertbildung kann durch ein Schieberegister als Tiefpassfilter gelöst werden. Insbesondere, wenn verschieden lange Schieberegister für Genauigkeit und kurze Gruppenlaufzeit kombiniert werden. Wie in Abb. 3.38 gezeigt, werden die letzten Bits des ausgegebenen Bitstroms gespeichert und akkumuliert. Für eine Obergrenze wird die Summe mit dem Wert verglichen, der den maximalen Ausgabewert darstellt ($-1 \leq \text{lim} \leq 1$). Überschreitet die Summe den Grenzwert, wird eine Null ausgegeben.

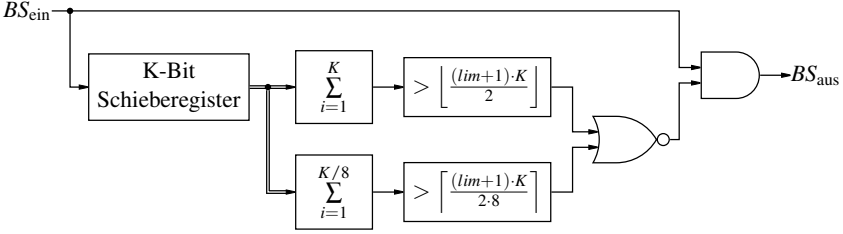


Abbildung 3.38: Positive Begrenzung eines Bitstroms basierend auf Schieberegistern [KS18].

Im Falle einer negativen Begrenzung wird die Summe mit dem Wert der Untergrenze verglichen. Unterschreitet die Summe den Grenzwert, wird eine Eins ausgegeben. Das resultierende Blockschaltbild ist in Abb. 3.39 dargestellt.

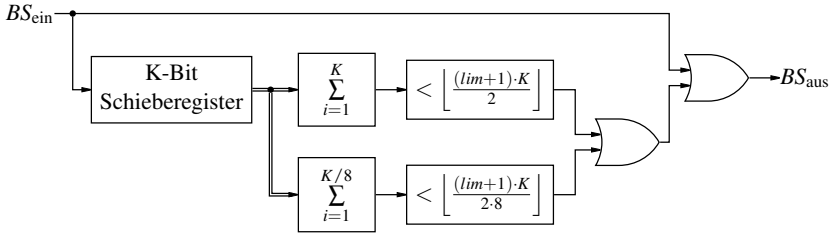


Abbildung 3.39: Negative Begrenzung eines Bitstroms basierend auf Schieberegistern [KS18].

Die Begrenzungen können bis zu einer Ober- und Untergrenze kaskadiert werden.

3.2.3.2 Ergebnisse

Die sinc^3 gefilterten Ergebnisse aus den Implementierungen für die Betriebsbegrenzung sind in Abb. 3.40 zu sehen. Wenn das Eingangssignal innerhalb des Grenzwertes bleibt, folgen die Ausgangssignale dem Referenzsignal. Die Implementierung auf Basis von Schieberegistern erzeugt während der Sättigung zusätzliches Rauschen und überschreitet dadurch

in einigen Punkten die Begrenzung. Die Varianz der DSV basierten Implementierung wird durch die Abweichungen zu Beginn und am Ende der Begrenzung verursacht. Diese ergibt sich aus den unterschiedlichen Filterlängen. Der kurze und damit schnelle sinc³-Filter berücksichtigt weniger Bits für den aktuellen Wert, sodass der gemittelte Wert bereits den Grenzwert erreicht, wobei der länger gefilterte Wert unter dem Grenzwert bleibt. Da der Mittelwert korrekt begrenzt ist, tritt dieser Effekt nur beim Betreten und Verlassen der Begrenzung auf. Die Implementierung auf Basis von Schieberegistern hat eine kürzere Gruppenlaufzeit als die auf Basis von DSV, mit 1 im Vergleich zu 15 Taktzyklen. Darüber hinaus zeigt die Implementierung auf der Grundlage von Schieberegistern eine kleinere Abweichung zum gewünschten Signal, wie in Tabelle 3.14 gezeigt.

Tabelle 3.14: Varianz und Gruppenlaufzeit der Operation Begrenzung [KS18].

Implementierung	Varianz	Gruppenlaufzeit
Schieberegister	8,7516e-6	1
DSV	1,3405e-5	15

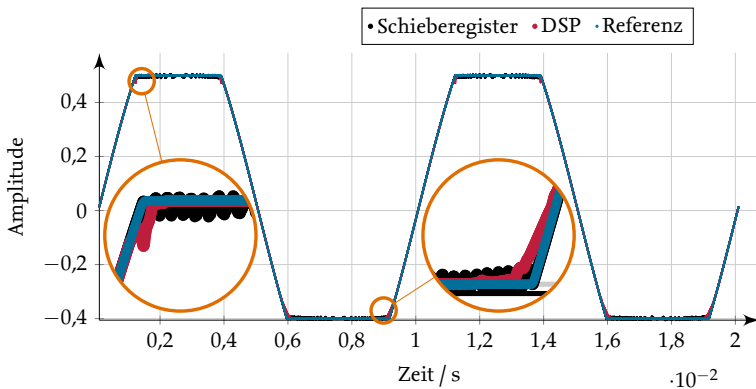


Abbildung 3.40: Tiefpass gefilterte Ergebnisse der Begrenzung mit einer Dezimierungsrate von 20 [KS18].

Daher wird in der Regel die Implementierung mit Schieberegistern empfohlen. Die auf DSV basierende Begrenzung sollte verwendet werden, wenn

eine benachbarte Operation zur Begrenzung auch in DSV implementiert ist. In diesem Fall wird keine zusätzliche Gruppenlaufzeit eingeführt, da die DSV Operationen kombiniert werden können, wodurch der zweite sinc^3 -Filter und der zweite $\Delta\Sigma$ -M weggelassen werden können. Ein weiterer Vorteil der Implementierung in DSV ist in Tabelle 3.15 dargestellt. Die Anzahl an benötigten Logikgattern der DSV Implementierung liegt bei 60 % der auf Schieberegister basierten Alternative. Mit einer anderen angrenzenden Operation, die auf DSV basiert, können die Implementierungskosten auf acht Logikgatter reduziert werden, die der Signalbreite an der Begrenzung entsprechen.

Tabelle 3.15: Ressourcenverbrauch der Operation Begrenzung auf einem FPGA [KS18].

Implementierung	Logikgatter
Schieberegister	303
DSV	181

3.3 Kaskadierte Operationen

Die Implementierung eines Regelgesetzes besteht meistens aus Kombinationen der Grundoperationen Skalierung, Summierung und Integration. Daher wird der SINAD-Verlauf in einer Kaskade von Operationen in der Simulation analysiert. Dazu regt eine Sinuswelle die Kaskade von $\Delta\Sigma\text{SV}$ Operationen mit einem Skalierungsfaktor von Eins an.

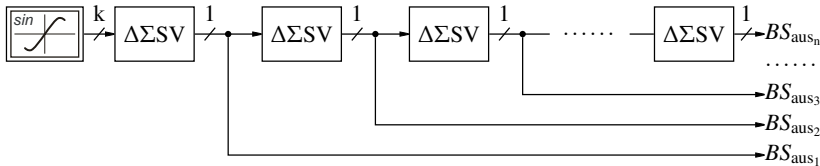


Abbildung 3.41: Versuch zur Evaluierung von kaskadierten $\Delta\Sigma\text{SV}$ Operationen.

Abb. 3.42 zeigt den SINAD für verschiedene $\Delta\Sigma$ -M Architekturen. Mit Ausnahme der Toolbox-Designs [33] mit zusätzlichen Nullstellen nimmt

das SINAD für eine zunehmende Anzahl von $\Delta\Sigma\text{SV}$ Operationen ab. Nach zehn $\Delta\Sigma\text{SV}$ Operationen entspricht der niedrigste SINAD immer noch einer effektiven Anzahl von 8,5 Bit. Eine Erhöhung der Ordnung des $\Delta\Sigma\text{-M}$ verbessert das SINAD signifikant. Dabei wird sie in Abb. 2.3 auf Seite 6 ohne resonante Rückführung zu Grunde gelegt. Die verwendeten Parameter der $\Delta\Sigma\text{-M}$ sind in Anhang A.1 abgedruckt.

Das Toolbox-Design mit zusätzlichen Nullstellen ist ein Sonderfall, da der SINAD konstant ist. Durch eine Signalübertragungsfunktion von 1 wird das Eingangssignal nicht verändert und jeder $\Delta\Sigma\text{-M}$ erzeugt das gleiche Quantisierungsrauschen bezogen auf das Eingangssignal. Die Bitströme des $\Delta\Sigma\text{-M}$ mit zusätzlichen Nullstellen sind somit identisch. Eine Simulation mit einem Verzögerungselement 1. Ordnung als $\Delta\Sigma\text{SV}$ Operation führt zu verschiedenen Bitströmen, deren SINAD mit einer Schwankung von unter 0,5 dB konstant ist. Diese Auslegung des $\Delta\Sigma\text{-M}$ ist nur in digitalen Implementierungen stabil, da die Koeffizienten $a_i = b_i$ und $b_{n+1} = 1$ gewählt werden. Daraus folgt, dass bei einer leichten positiven Abweichung von einem b_i der $\Delta\Sigma\text{-M}$ instabil wird.

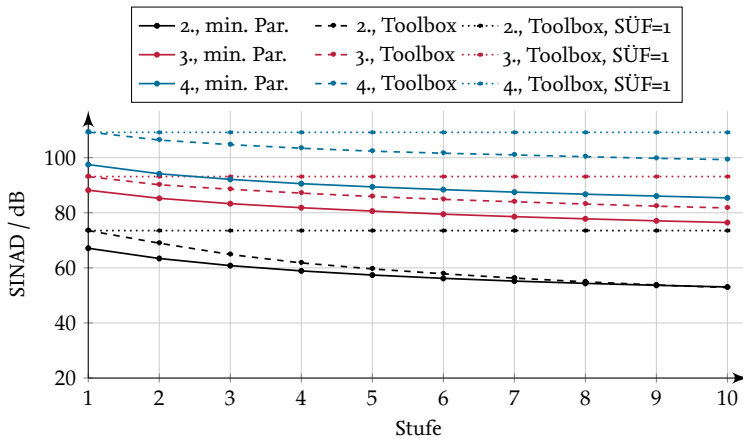


Abbildung 3.42: Signalqualität von kaskadierten $\Delta\Sigma\text{SV}$ Operationen [HKS16].

Mit den bisherigen Ergebnissen ist es möglich, das Regelgesetz direkt auf den Bitströmen der Messdaten zu berechnen. Um den Sollwert über eine

moderne Endstufe auszugeben, muss eine mittlere Schaltfrequenz $\leq 40 \text{ kHz}$ erzielt werden. Dafür wird im nächsten Kapitel eine Implementierung für dreiphasige Lasten vorgestellt, die $\Delta\Sigma$ -PWM.

4 Delta-Sigma PWM Modulator

In [10, 11] wird eine Lösung zur Umwandlung der hochfrequenten Bitströme in Steuersignale für Leistungselektronik vorgestellt. Der $\Delta\Sigma$ -PWM ist der Modulator, der im Gegensatz zu anderen Modulatoren [4, 24], eine Raumvektor-Schaltsequenz mit beiden Nullvektoren wie in der klassischen Raumzeigermodulation (RZM) Sequenz aufweist. Dieser Ansatz sorgt für gleiche Schaltfrequenzen auf allen Phasen und damit für eine symmetrische thermische Belastung des Umrichters und der angeschlossenen Last.

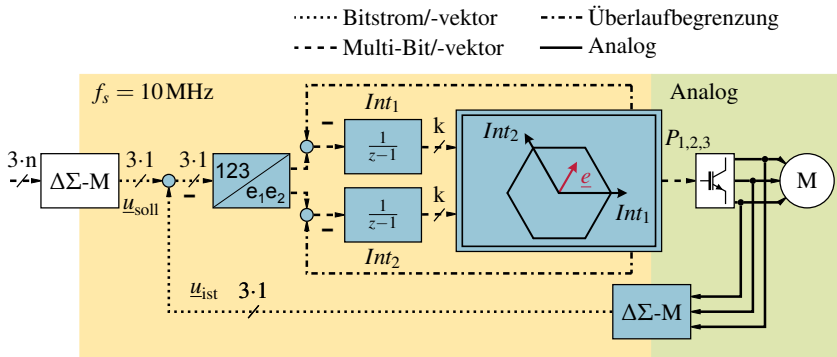


Abbildung 4.1: $\Delta\Sigma$ -PWM mit interner [Spannungsregelung](#) [10].

Der sogenannte $\Delta\Sigma$ -PWM, dargestellt in Abb. 4.1, besteht aus einem Integratorpaar, das den Spannungsregelungsfehler akkumuliert und basierend auf dem zweidimensionalen Regelfehler hysteresebasierte Schaltvorgänge durchführt. Durch die Berechnung aller Operationen in $\Delta\Sigma$ SV ermöglicht die Struktur die bestmögliche Dynamik [10]. Der Spannungsregelfehler e wird in einem 120° System berechnet. Dies wird aus dem dreiphasigen System gebildet, indem die dritte Phase von den anderen beiden abgezogen wird. Das zweidimensionale Hystereseschaltelement ist die zentrale

Komponente. Es kombiniert Betrag und Phasenhysterese, wie in Abb. 4.2 gezeigt. Für die Betragshysterese wird die Vereinfachung aus [KTS18b] verwendet. Dafür wird die Hysteresegrenze N_0 weggelassen, was sich nicht negativ auf das Schaltverhalten der $\Delta\Sigma$ -PWM auswirkt. Die beiden Betragshysteresen N_1 und N_2 bleiben erhalten. Die Grenze N_1 beeinflusst die Schaltfrequenz, da ein Schaltereignis durch Überschreiten ausgelöst wird. Die N_2 Grenze wird verwendet, um Übersteuerungen und damit das Überlaufen der Integratoren zu erkennen. N_2 sollte im Normalbetrieb nicht erreicht werden. Die Überschreitung des Regelfehlers aufgrund von Totzeiten (t_d) und Messverzögerung (t_m) muss für die Auslegung des Offsets C zwischen N_1 und N_2 , wie in [KTS18b] dargestellt, berücksichtigt werden. Wie dort angegeben, sollte er in folgendem Bereich liegen

$$C = 1 \dots 2 \cdot \sqrt{3} \cdot (t_d + t_m). \quad (4.1)$$

Im Falle von Übermodulation, wenn der Spannungsfehler außerhalb des N_2 Sechsecks liegt, werden die Integratoren des Spannungsregelungsfehlers parallel zum aktiven Ausgangsvektor reduziert.

Die Phasenhysterese teilt sich in zwei Sätze auf, einer ist für V_1 Vektoren gültig, siehe Abb. 4.2(b) und der andere für V_2 Vektoren, siehe Abb. 4.2(c). Somit wird jedem aktiven Schaltvektor ein 120° Hysteresesektor zugeordnet, was in 60° Hysterese zu beiden benachbarten Vektoren resultiert. Diese zwei Sätze Phasenhysterese vermeiden hochfrequente Schaltzyklen.

Basierend auf dem Betrags- und Phasenverlauf des Regelfehlers erzwingt eine Zustandsmaschine die RZM Sequenz

$$V_{00} \rightarrow V_1 \rightarrow V_2 \rightarrow V_{01} \rightarrow V_2 \rightarrow V_1 \rightarrow V_{00}, \quad (4.2)$$

die bei jeder Transition genau eine Phase umschaltet.

Der interne Regelkreis kann auf zwei Arten geschlossen werden, zum Einen durch die Rückführung der leistungselektronischen Gatesignale, zum Anderen durch die gemessenen Phasenspannungen. Die Rückführung der Gatesignale kann verwendet werden, wenn Leistungselektronik mit kleinem Spannungsabfall und kurzen Totzeiten, zum Beispiel Metall-Oxid-Halbleiter-Feldeffekttransistor (MOSFET) verwendet werden. Bei

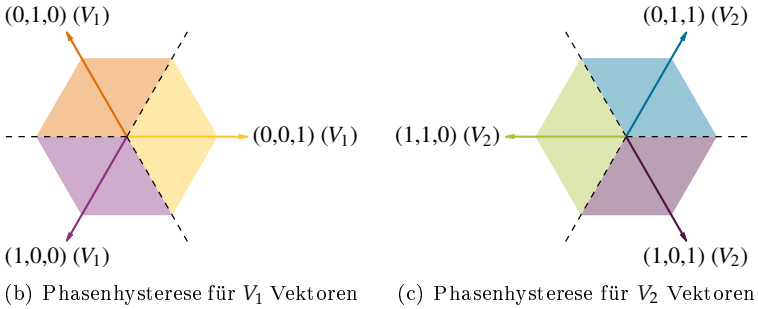
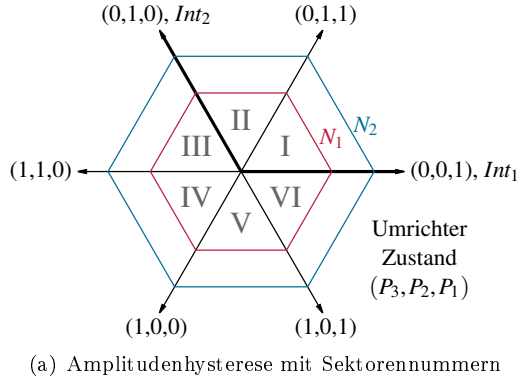


Abbildung 4.2: Amplituden- und Phasenhysterese des $\Delta\Sigma$ -PWM [KTS18b].

einer variablen Zwischenkreisspannung oder für Leistungselektronik mit signifikantem Spannungsabfall und größeren Totzeiten, zum Beispiel Bipolartransistor mit isolierter Gate-Elektrode (IGBT) empfiehlt sich die Rückführung der Phasenspannungen, wie in Abb. 4.1 dargestellt. Die resultierende Kontrollstruktur bildet den Hauptteil des $\Delta\Sigma$ -PWM, der blau eingefärbt ist.

4.1 Prüfstand

Um die diskutierten und simulierten Ergebnisse zu verifizieren, wird der in [26] eingeführte Prüfstand mit der in [TKHS17] vorgestellten Version mit $\Delta\Sigma$ -M als DAU verwendet. Es basiert auf einem FPGA, das alle

Regelfunktionen mit einem internen Taktzyklus von 100 MHz ausführt, wodurch der $\Delta\Sigma$ -PWM sowie andere $\Delta\Sigma$ SV-Algorithmen mit einer Abtastrate von 10 MHz ausgeführt werden können. Zur Auswertung der Daten im 10 MHz Raster werden Messdaten in einem synchronem dynamischem Speicher mit wahlfreiem Zugriff (SDRAM) aufgezeichnet [HKKS16]. Die Messungen wurden auf dem Prüfstand mit einem Permanentmagnet-Synchronmotor (PMSM) durchgeführt. Die Motorparameter sind in Tabelle 4.1 abgedruckt.

Tabelle 4.1: Motordaten [6]

Parameter	Einheit	Wert
Anzahl der Pole		8
Anzahl der Phasen		3
Nenn-Leistung	W	100
Nenn-Spannung	V	24
Nenn-Strom	A	6,95
Pol zu Pol Widerstand	Ω	0,28
Pol zu Pol Induktivität	mH	0,54

4.2 Operationsmodi

Der folgende Abschnitt beschreibt die verschiedenen Betriebsarten des $\Delta\Sigma$ -PWM und basiert auf [KTHS17]. Die Betriebsart beeinflusst hauptsächlich die Modulationsstrategie.

Der $\Delta\Sigma$ -PWM kann die aktuelle Betriebsart nahtlos zwischen RZM, Auslassen von Nullvektoren (ANV), Übermodulation (ÜM) und Übermodulation mit Überlaufbegrenzung (ÜMÜ) umschalten. Basierend auf der aktuellen Zwischenkreisspannungsausnutzung wählt der $\Delta\Sigma$ -PWM die geeignete Betriebsart, wobei er möglichst RZM betreibt. Am Beispiel eines regulären RZM Zyklus, wie in Abb. 4.3 auf Seite 66 gezeigt, wird die Grundfunktionsweise der $\Delta\Sigma$ -PWM erläutert. Für den Zyklus wird ein statischer Sollwert \underline{U}_{set} im Sektor I angenommen. Dadurch ergibt sich, dass Int_1 der dominante und Int_2 der rezessive Integrator ist, da die Projektion des Sollwerts in Int_1 Richtung größer ist als in Int_2 Richtung. Des Weiteren

wird von idealen Schaltern und verzögerungsfreien Messungen ausgegangen und angenommen, dass das System sich im eingeschwungenen Zustand befindet. Der integrale Regelfehler beginnt am Ende des vorausgegangenen PWM-Zyklus, der Sektorgrenze, bei der das Kriterium $Int_1 = Int_2$ erfüllt ist ($Int_1(0) = Int_1$, $\underline{e}(0) = (Int_1(0), Int_1(0))$) und der Modulator V_{0_0} ausgibt. Zusätzlich wird das Zyklusbit, das die Halbzyklen identifiziert, auf Null gesetzt. $Int_1(0)$ ist ein vom Sollwert abhängiger Anfangswert des Schaltzyklus. Bei einem aktiven Nullvektor bewegt sich \underline{e} in Richtung U_{set} . Das erste Schaltereignis wird ausgelöst, wenn $|\underline{e}| \geq N_1$. Für den angegebenen Sollwert ist das der Fall, wenn $Int_1 = N_1$. Der Punkt des Schaltereignisses ist als kleiner Kreis in Abb. 4.3 dargestellt. Wenn das Schaltereignis eintritt, schaltet der Modulator auf den Vektor V_1 entsprechend der Phase von \underline{e} nach Abb. 4.2(b) auf Seite 63, hier (001). Der kumulierte Fehler bewegt sich in Richtung des Vektors $U_{set} - V_1$. Das nächste Schaltereignis wird ausgelöst, sobald die Fehlertrajektorie die dem aktiven Vektor V_1 zugeordnete Winkelhysterese verlässt. Im angegebenen Beispiel ist dies das Ereignis, wenn $Int_1 = Int_2$. In diesem Fall wird der die Grenze bildende Vektor V_2 (011) aktiviert, da der Regelfehler sich nach Abb. 4.2(c) auf Seite 63 in der entsprechenden Phasenhysterese befindet. Das letzte Schaltereignis des ersten Halbzyklus wird ausgelöst, wenn der Regelfehler die dem aktiven Vektor zugeordnete Phasenhysterese verletzt. Für den untersuchten Fall wird das Schaltereignis ausgelöst, wenn $Int_2 = 0$. In diesem Fall wird V_{0_1} als Ausgangsvektor geschaltet. Der zweite RZM Halbzyklus beginnt und daher wird das Zyklusbit auf eins gesetzt. Laut Gleichung (4.2) ist der nächste aktive Vektor ein V_2 -Vektor mit entsprechender Phasenhysterese, dargestellt in Abb. 4.2(c) auf Seite 63. Im beschriebenen Szenario wird der Vektor V_2 (011) ausgewählt. Er wird aktiviert, wenn $|\underline{e}| \geq N_1$. Wie zuvor beschrieben, entspricht dies $Int_1 = N_1$. Da der zweite Halbzyklus dem gespiegelten ersten Halbzyklus entspricht, führt das nächste Schaltereignis zu einem V_1 -Vektor, (001) im gezeigten Fall und wird durch Phasenhysterese ausgelöst. Dies geschieht bei $Int_2 = 0$. Um einen Raumvektorzyklus abzuschließen, tritt das letzte Schaltereignis auf V_{0_0} ein, wenn die Phasenhysterese gemäß Abb. 4.2(b) erreicht ist. Dies entspricht dem

neue Indikatoren notwendig. Diese untersuchen einen vollständigen RZM Zyklus, begrenzt durch den Übergang des Zyklusbits von Eins zu Null. Wird die Amplitudenhysterese N_2 in einem Zyklus erreicht ($N_{2,c}$), arbeitet der Modulator in ÜMÜ. Zur Auswertung wird ein Indikatorbit (b_{N2}) gesetzt, das am Ende des RZM Zyklus ausgewertet und zurückgesetzt wird. Wenn die Amplitudenhysterese N_2 nicht erreicht wird und das zweite und fünfte Schalteignis von Gleichung (4.2) zwischen N_1 und N_2 in einem RZM Zyklus liegen, befindet sich der Modulator im ÜM Modus. Liegt nur eines der beiden Schalteignisse zwischen N_1 und N_2 in einem RZM Zyklus, ist der $\Delta\Sigma$ -PWM in der Betriebsart ANV. In dieser wird einer der beiden Nullvektoren ausgelassen. Der Indikator unterscheidet nicht zwischen dem Auslassen von V_{00} und von V_{01} , was bedeutet, dass der Modus Flatbottom- und Flattop-Schaltzyklen [9, 36] umfasst. Um den Zustand des Modulators anzuzeigen, wird am Ende eines RZM Zyklus die Anzahl der verwendeten Nullvektoren ausgewertet. Zwei aktivierte Nullvektoren indiziert RZM, ein aktivierter Nullvektor indiziert ANV und null aktivierte Nullvektoren indizieren ÜM. Um zwischen ÜM und ÜMÜ zu unterscheiden, wird b_{N2} verwendet.

Abb. 4.4 zeigt das relative Auftreten der erwähnten Indikatoren für verschiedene m . Die Indikatoren für ÜM und ANV sind bewusst nicht abgedruckt, da sie in der Simulation mit echten Schaltern Null sind. Dieses Verhalten wird durch die externe Totzeitlogik verursacht. Diese führt zu einer verzögerten Ausgabe des Einschaltsignals für die Leistungselektronik, um Kurzschlussströme im Wechselrichter zu verhindern. Impulse, deren Dauer kleiner als die Totzeit sind, entfallen. Für den $\Delta\Sigma$ -PWM wird die RZM Sequenz nicht verletzt und somit ist er noch im RZM Modus. Die integrierte Spannungsregelung des $\Delta\Sigma$ -PWM kompensiert die Abweichung der aktiven Spannungs-Zeitflächen. Aufgrund der Filterung der Steuersignale durch die Totzeitlogik sind die möglichen Einsparungen an Spannungszeitflächenverlusten bereits ausgeschöpft und führen zu keiner weiteren Verbesserung der Spannungsausnutzung. Für den Fall, dass der Sollwert die Grenze überschreitet, bei der die Operationsmodi ANV und ÜM ausreichen, um den Sollwert an der Last umzusetzen, erkennt der Modulator die Abweichung und ändert den Betriebsmodus auf ÜMÜ. Somit ist der

Indikator für ÜMÜ ein scharfes Indiz für die Überschreitung der vom Zwischenkreis verfügbaren Spannungsgrenze.

Die Anzahl der Taktzyklen außerhalb der Amplitudenhysterese N_1 geteilt durch die Taktzyklen der Raumvektorsequenz ergibt einen nichtlinearen Spannungsausnutzungsindikator $|\underline{e}| > N_1$. Der vorgeschlagene N_{1X} Indikator [10] zeigt ein von der Einstellung abhängiges Verhalten zwischen dem Verhalten des $|\underline{e}| > N_1$ und dem $N_{2,c}$ Indikators. Aufgrund des Ausbleibens der Identifikatoren für ÜM und ANV wird als Spannungsausnutzungsindikator $|\underline{e}| > N_1$ vorgeschlagen.

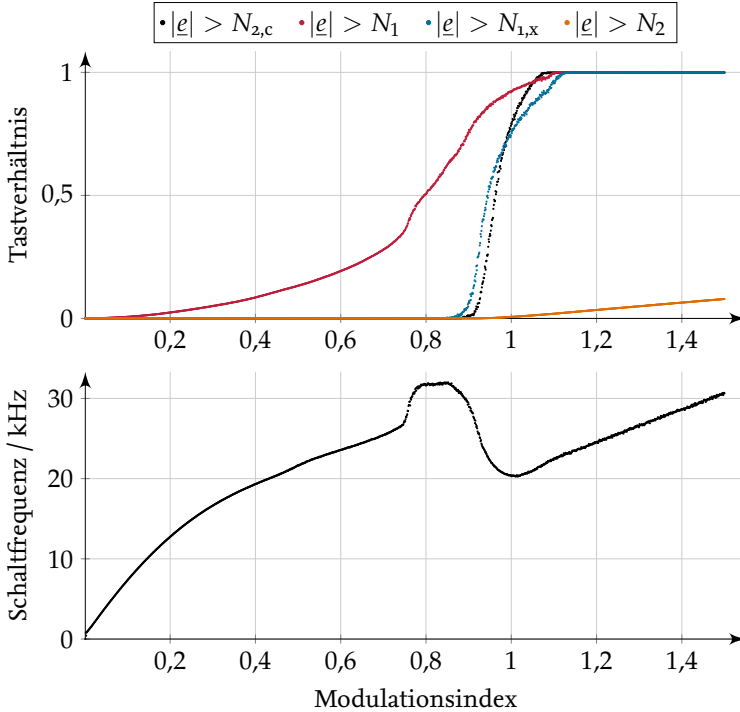


Abbildung 4.4: Online Indikatoren für Betriebsart- oder Lastpunkterkennung über m aus der Simulation. $\varphi_{soll} = 30^\circ$, $|\underline{e}| > N_{2,c}$ mindestens einmal pro SVM-Zyklus [KTHS17].

Die Ergebnisse führen zu dem neuen Indikator $|\underline{e}| > N_1$ für eine dynamische Abschätzung der tatsächlichen Spannungsausnutzung, die für eine geregelte Zwischenkreisspannung oder einen Feldschwächungsregler not-

wendig ist. Das Verhältnis zwischen $|\underline{e}| > N_1$ und der Betriebsart des $\Delta\Sigma$ -PWM hängt von leistungselektronischen Parametern wie Totzeiten und den $\Delta\Sigma$ -PWM Amplitudenhysteresen ab. Der Indikator $N_{2,c}$ dient zur scharfen Erkennung der Zwischenkreisauslastung und zeigt unabhängig von Leistungselektronik und Modulatorparametern an, dass ÜMÜ verwendet wird. Darüber hinaus haben diese Indikatoren den Vorteil, dass sie von Null bis Eins skaliert sind.

4.3 Schaltfrequenz

Der folgende Abschnitt beschreibt eine Möglichkeit die Schaltfrequenz analytisch zu berechnen, wie in [KTHS17] dargestellt und die natürliche Schaltfrequenz des $\Delta\Sigma$ -PWM wird vorgestellt.

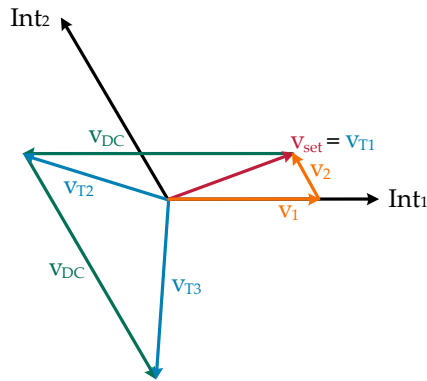


Abbildung 4.5: Erwartete mittlere Geschwindigkeit des Regelfehlers bei unterschiedlichen Schaltzuständen [KTHS17].

Die Schalthandlungen des $\Delta\Sigma$ -PWM sind hysteresebasiert. Daraus folgt, dass die aktuelle Schaltfrequenz vom Arbeitspunkt abhängig ist. Aus der integralen Regelfehlertrajektorie in Abb. 4.3 auf Seite 66, die in Abschnitt 4.2 auf Seite 64 erläutert ist, kann mit den Geschwindigkeiten

des integralen Regelfehlers aus Abb. 4.5 ein lineares Gleichungssystem zur Bestimmung der Raumzeigermodulationszykluszeit aufgestellt werden

$$\begin{bmatrix}
 v_1 & 0 & 0 & 0 & 0 & 0 & 1 \\
 v_2 & v_{DC} - v_1 + v_2 & 0 & 0 & 0 & 0 & 1 \\
 v_2 & v_2 & v_2 - v_{DC} & 0 & 0 & 0 & 1 \\
 v_2 & v_2 & v_1 - v_{DC} & v_1 & 0 & 0 & 1 \\
 0 & 0 & 0 & v_2 & v_2 - v_{DC} & 0 & 0 \\
 0 & 0 & 0 & 0 & v_{DC} - v_1 & v_{DC} - v_1 + v_2 & 0 \\
 0 & 0 & 0 & 0 & 0 & v_2 & -1
 \end{bmatrix} \cdot \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \\ T_6 \\ Int_1(0) \end{bmatrix} = \begin{bmatrix} N_1 \\ N_1 \\ 0 \\ N_1 \\ 0 \\ N_1 \\ 0 \end{bmatrix} . \quad (4.3)$$

Dabei sind die Geschwindigkeiten (v_x) in Inkrementen pro Abtastschritt, die Zeiten (T_i) in Abtastschritten und die Hysteresebreite (N_1), sowie der Startwert ($Int_1(0)$) in Inkrementen angegeben.

Die Lösung von Gleichung (4.3) enthält zwei Informationen, den Startwert von $\underline{e} = (Int_1(0), Int_1(0))$ und die Zeiten der einzelnen Schaltzustände (T_1 - T_6). Für einen konstanten Sollwert und eine konstante Zwischenkreisspannung wird $Int_1(0)$ konstant. Das heißt, dass der Sollwert exakt erreicht wird, da die Ableitung des integralen Regelfehlers, der Regelfehler, über einen PWM-Zyklus zu Null wird.

Aus den Zeiten der Schaltzustände und der Abtastfrequenz (f_s) lässt sich die Schaltfrequenz berechnen

$$f_{\text{PWM}} = \left(\sum_{i=1}^6 T_i \right)^{-1} \cdot f_s. \quad (4.4)$$

Die algebraische Lösung ist in Gleichung (A.19) auf Seite 112 dargestellt. Die Geschwindigkeiten des integralen Regelfehlers (v_1, v_2) ergeben sich

aus dem aktuellen Sollwert und der maximalen Aussteuerung der digitalen $\Delta\Sigma$ -M der Sollspannungen, die Geschwindigkeit v_{DC} ist abhängig von der Zwischenkreisspannung (u_{DC}) und der maximal messbaren Spannung

$$v_1 = \left(\cos(\varphi) + \frac{1}{\sqrt{3}} \sin(\varphi) \right) \frac{|u_{\text{set}}|}{u_{\Delta\Sigma}}, \quad (4.5)$$

$$v_2 = \frac{2}{\sqrt{3}} \sin(\varphi) \frac{|u_{\text{set}}|}{u_{\Delta\Sigma}}, \quad (4.6)$$

$$v_{DC} = \frac{2 \cdot u_{DC}}{u_{\Delta\Sigma}}. \quad (4.7)$$

Allgemein ist es sinnvoll, die maximale Aussteuerung der digitalen $\Delta\Sigma$ -M der Sollspannungen gleich der maximalen zu messenden Spannung zu setzen, daher werden beide Größen mit $u_{\Delta\Sigma}$ bezeichnet.

Aus der algebraischen Lösung lassen sich Vorhersagen zur Schaltfrequenz berechnen, falls der Spannungssollwert als Multibit-Wert vorliegt, was in der Regel nicht der Fall ist. Weiter ist der Rechenaufwand für eine online Berechnung sehr hoch. Somit eignet sich das Verfahren nicht zur Schaltfrequenzregelung.

Der Schaltfrequenzverlauf eines realen Systems mit dreiphasiger sinusförmiger Spannungsanregung und steigendem Modulationsindex ist in Abb. 4.6 dargestellt. Dieser variiert zwischen 2 kHz und 14 kHz, daher wird ein online berechenbarer Ansatz für die Schaltfrequenzregelung benötigt.

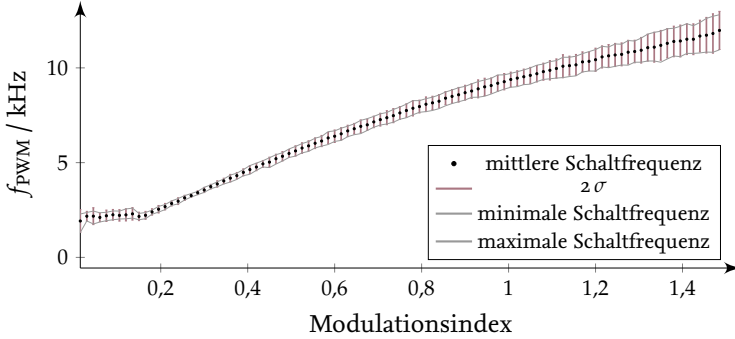


Abbildung 4.6: Natürliche Schaltfrequenz des $\Delta\Sigma$ -PWM über der Aussteuerung [KTS18b].

4.4 Schaltfrequenzregelung

Insbesondere die niedrigen Schaltfrequenzen stellen eine Herausforderung für den Zwischenkreis und gegebenenfalls eingesetzter Bootstrap-Schaltung dar [35]. Ein Absenken der Schaltfrequenz in bestimmten Arbeitspunkten der $\Delta\Sigma$ -PWM erfordert größere Zwischenkreisfilter und Bootstrap-Kondensatoren im Vergleich zu einer Festfrequenz-PWM mit der gleichen durchschnittlichen Schaltfrequenz. Um die benötigten Kapazitäten zu reduzieren, wird ein Regler für die Schaltfrequenz des $\Delta\Sigma$ -PWM eingeführt. Dieser ist in [KTS18b] für die Spannungsregelung vorgestellt. In dieser Arbeit wird der Schaltfrequenzverlauf der $\Delta\Sigma$ -PWM ohne äußere Reglerkaskade betrachtet. Der Schaltfrequenzregler passt die Amplitudenhysterese N_1 an, um die Schaltfrequenz auf Basis der Dauer des letzten PWM-Halbzyklus und dem vorherigen Wert von N_1 zu regeln ($N_{1_{\text{alt}}}$). Dabei ist das Verhältnis von N_1 zu $N_{1_{\text{alt}}}$ gleich dem Verhältnis der gewünschten (t_{soll}) zur benötigten Halbzyklusdauer (t_{ist})

$$\frac{N_1}{N_{1_{\text{alt}}}} = \frac{t_{\text{soll}}}{t_{\text{ist}}}. \quad (4.8)$$

Die Zeitmessung in einem FPGA basiert auf dem Zählen von Taktzyklen, daher wird im Folgenden die auf die Abtastfrequenz normierte Zeit

$T = t \cdot f_s$ verwendet. In Gleichung (4.8) kürzen sich die Normierungen heraus, sodass t durch T ersetzt werden kann.

Die gewünschte Halbzyklusdauer ergibt sich dabei aus der Hälfte der Abtastfrequenz (f_s) geteilt durch die gewünschte Schaltfrequenz (f_{soll})

$$T_{\text{soll}} = \frac{f_s}{2 f_{\text{soll}}}. \quad (4.9)$$

Die aktuelle Halbzyklusdauer ergibt sich aus der Anzahl der Taktzyklen, die vergangen sind, während der Modulator die ersten drei oder die zweiten drei Schaltzustände ausgegeben hat

$$T_{\text{ist}} = \sum_{i=1|4}^{3|6} T_i, \quad (4.10)$$

wobei $a|b$ entweder als a oder b interpretiert wird, sodass die Summe über $i = 1 \dots 3$ oder $i = 4 \dots 6$ gebildet wird.

Gleichung (4.8) bis Gleichung (4.10) ergeben zusammengefasst eine Gleichung zur Berechnung von N_1

$$N_1 = \frac{N_{1\text{alt}}}{\sum_{i=1|4}^{3|6} T_i} \cdot \frac{f_s}{2 f_{\text{soll}}}, \quad (4.11)$$

mit der die Hystereseschwelle berechnet wird, mit welcher der letzte PWM-Halbzyklus die eingestellte PWM-Frequenz erreicht hätte. Da die PWM-Frequenz höher sein sollte als die höchste Frequenz im modulierten Signal, gilt die Annahme nahezu konstanter Eingangswerte [35], was einem großen Abstand zwischen der Signalfrequenz und der Schaltfrequenz bedeutet. So wird der nächste PWM-Zyklus eine ähnliche Zeit in Anspruch nehmen, besonders bei kleinen eingestellten Amplituden [KTHS17]. Das Blockschaltbild des Frequenzreglers ist in Abb. 4.7 dargestellt.

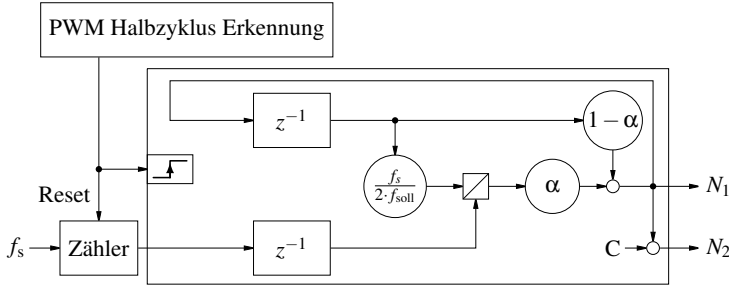


Abbildung 4.7: Schaltfrequenzregelung des $\Delta\Sigma$ -PWM mit Filter [KTS18b].

Der optionale unendliche Impulsantwortfilter (IIR-Filter), der durch die Faktoren α und $1 - \alpha$, mit $\alpha \in [0 \dots 1]$ implementiert ist, kann verwendet werden, um die Hystereseeänderung zu glätten. Mit dem Parameter $\alpha = 1$ wird der Filter deaktiviert.

Bei Modulationsindizes nahe und oberhalb der Spannungsgrenze ist der Winkel des Sollwertes der Haupteinfluss auf die tatsächliche Schaltfrequenz [KTHS17], siehe Abb. 4.8. Bei Sollwerten an der Spannungsgrenze und höher degeneriert die Raumzeiger-Modulationssequenz und hat nur einen Schaltzustand pro Halbzyklus. Dies muss in der Schaltfrequenzregelung berücksichtigt werden, da die Annahme gleicher Halbwertszeiten bei hohen Aussteuergraden nur für Winkel gilt, bei denen der Sollwert in Richtung Int_1 und Int_2 gleich groß ist. Daher wird in diesen Fällen die volle PWM-Zyklusdauer geteilt durch zwei als Sollgröße für die Frequenzregelung verwendet. Da der Drehwinkel des Sollwertes zu einer hochfrequenten Störung der tatsächlichen Schaltfrequenz führt, die nicht gesteuert werden kann und soll, wird der optionale Filter durch Setzen von $\alpha = 0,8$ aktiviert.

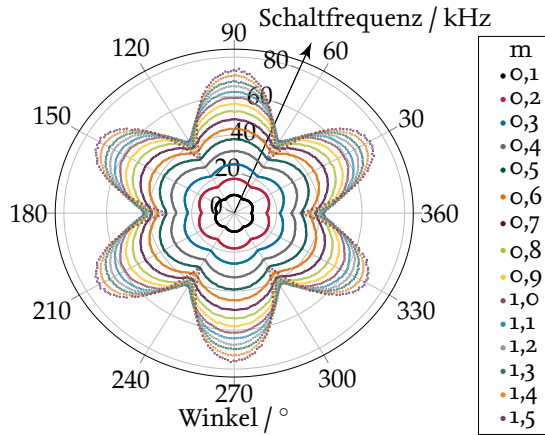


Abbildung 4.8: Schaltfrequenzen des $\Delta\Sigma$ -PWM bei unterschiedlichen Aussteuerungen [KTHS17].

Wie in Abb. 4.9 dargestellt, ergibt die durchschnittliche Frequenzregelung eine fast gaußsche Verteilung. Bei kleinen eingestellten Amplituden existiert eine große statistische Streuung, die bei Modulationsindizes über 0,5 signifikant abnimmt. Die gewünschte mittlere Schaltfrequenz kann auch erst ab diesem Modulationsgrad stabil geregelt werden.

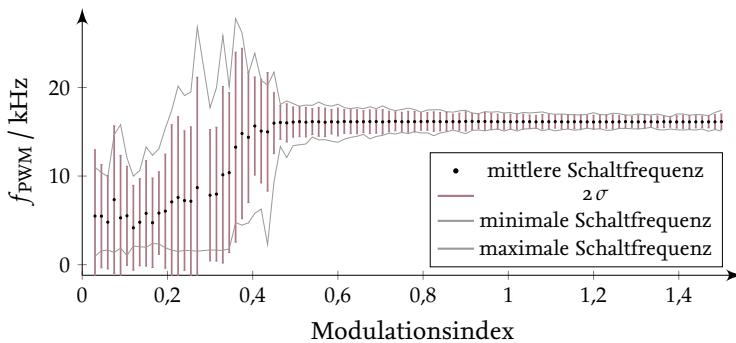


Abbildung 4.9: Geregelte Schaltfrequenz (16 kHz) des $\Delta\Sigma$ -PWM über der Aussteuerung [KTS18b].

Wie am Anfang dieses Abschnitts erwähnt, sollte die minimale Schaltfrequenz begrenzt werden. Aufgrund der Auswertung des letzten Schaltzyklus und Berechnung des neuen Sollwertes für die nächste PWM-Periode kann der vorgestellte Frequenzregler keinen Einfluss auf die aktuelle PWM-Periode nehmen. Eine zusätzliche Schaltung, der minimale Frequenzwächter, kann im aktuellen PWM-Zyklus Schalthandlungen erzwingen. Dazu überwacht er die Zeit in Taktzyklen der aktuellen Schalthandlung. Überschreitet diese die Hälfte der gesamten Zykluszeit, wird eine Schalthandlung ausgelöst. Das Blockschaltbild ist in Abb. 4.10 dargestellt.

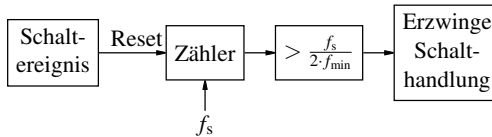


Abbildung 4.10: Blockschaltbild des minimalen Schaltfrequenzwächters [KTS18b].

In dem Fall einer erzwungenen Schalthandlung wird entsprechend des aktuellen Zustands, definiert durch den Schaltvektor, den Regelfehler und das Zyklusbit der nächste Ausgangsvektor ausgewählt. Somit ergibt sich eine untere Schranke für die Dauer eines PWM-Zyklus.

Die Grundidee des minimalen Schaltfrequenzwächters besteht darin, den PWM-Zyklus in zwei Halbzyklen mit jeweils drei Ausgangszuständen (Nullvektor, V1-Vektor und V2-Vektor) aufzuteilen. Der minimale Schaltfrequenzwächter sorgt somit für eine Schaltfrequenz zwischen einem Drittel und Eins der Minimalschaltfrequenz. Ein Extremfall tritt ein, wenn ein Vektor dominant ist und die aktive Zeit des dominanten Vektors begrenzt werden muss. Die resultierende Schaltfrequenz entspricht der eingestellten minimalen Schaltfrequenz. Der andere Extremfall resultiert, wenn sich der Sollwert während eines PWM-Halbzyklus derart ändert, dass alle drei Ausgangsvektoren zeitlich begrenzt werden müssen. In diesem Fall ergibt sich die Schaltfrequenz zu einem Drittel der eingestellten minimalen Schaltfrequenz.

Wie in Abb. 4.11 ersichtlich ist, sinkt die resultierende Schaltfrequenz etwa linear mit steigendem Sollwert bis auf die Hälfte der inversen maximalen Haltezeit im unteren Amplitudenbereich. Bei hohen Sollwerten konvergiert die Schaltfrequenz auf $2/3$, da nur die beiden aktiven Vektoren verwendet werden.

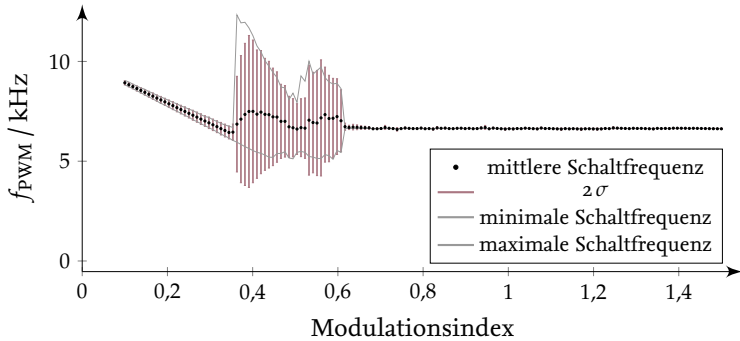


Abbildung 4.11: Schaltfrequenz mit Schaltfrequenzwächter (10 kHz) des $\Delta\Sigma$ -PWM über der Aussteuerung [KTS18b].

Bei einem durch den minimalen Schaltfrequenzwächter verursachten Schaltvorgang muss der Frequenzregler sein Regelgesetz ändern. Der neue Wert von N_1 wird aus dem maximalen Fehlerbetrag ($|\underline{e}|_{\max}$, in Inkrementen) im letzten PWM-Halbzyklus und den Taktzyklen des letzten PWM-Halbzyklus berechnet. Da die resultierende Grenze für N_1 die Schaltfrequenz f_{\min} bewirkt, wird sie nach [KTHS17] für f_{soll} skaliert. Die Berechnung von N_1 folgt somit

$$N_1 = 2 \cdot |\underline{e}|_{\max} \cdot \sum_{i=1|4}^{3|6} T_i \cdot \frac{f_{\text{soll}}}{f_s}. \quad (4.12)$$

Die resultierende PWM-Schaltfrequenz bei der Verwendung aller vorgestellten Regler ergibt die Verteilung in Abb. 4.12 über dem Lastpunkt. Im Vergleich zu Abb. 4.9 auf Seite 75 ist die mittlere Schaltfrequenz auf die eingestellte Minimalfrequenz begrenzt. Wie zu erwarten, treten geringere

Schaltfrequenzen auf. Das stellt in der Anwendung kein Problem dar, da sie unter 10 % abweichen.

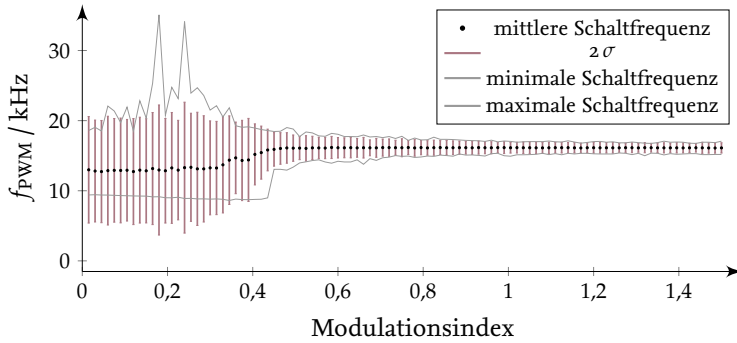


Abbildung 4.12: Geregelte Schaltfrequenz (16 kHz) mit Schaltfrequenzwächter (10 kHz) des $\Delta\Sigma$ -PWM über der Aussteuerung [KTS18b].

5 Anwendungen

Allgemein kann der im vorherigen Kapitel vorgestellte $\Delta\Sigma$ -PWM für die Ansteuerung von dreiphasigen Wechselrichtern verwendet werden. Im Folgenden Kapitel werden die Einsatzmöglichkeiten des $\Delta\Sigma$ -PWM im Bereich der elektrischen Antriebsregelung vorgestellt und evaluiert. Dabei werden die Messgrößen mit $\Delta\Sigma$ -M aufgenommen und mittels $\Delta\Sigma$ SV weiter verarbeitet.

Bei den Messungen ist zu beachten, dass die analogen Signale kontinuierlich den zu messenden Wert führen, da sie mit der Abtastfrequenz der $\Delta\Sigma$ -M ausgewertet werden und der Mittelwert der Bitströme die Signale repräsentiert. Im Bereich der Antriebsregelung sind zum Beispiel Widerstände zur Strommessung im unteren Wechselrichterzweig nicht geeignet, da hier die Messgröße nur in bestimmten Schaltzuständen gültig ist.

Der Modulationsindex (m) bezieht sich auf die maximale Stranggröße im Einphasensystem.

5.1 Stromregelung

In der Antriebsregelung werden bei Raumzeigermodulation typischerweise Stromregler verwendet. Die vorgestellte $\Delta\Sigma$ -PWM realisiert eine hochdynamische Spannungsregelung. Gemäß der Regelungstheorie kann die Stromregelung als äußerer Regelkreis überlagert werden [22]. Die Kaskadenstruktur stellt einen großen Vorteil des $\Delta\Sigma$ -PWM dar. In der Umsetzung mittels symmetrischer PWM wird auf eine Spannungsregelung verzichtet [36], da in einer Kaskadenregelung die Bandbreite von der innersten zur äußersten Kaskade abnimmt. Somit müsste die PWM mit einer höheren Schaltfrequenz betrieben werden, um die gleiche Dynamik im Stromregelkreis mit unterlagelter Spannungsregelung zu erzielen, wie der $\Delta\Sigma$ -PWM.

5.1.1 Proportional Stromregler im dreiphasigen System

Der klassische Ansatz für die $\Delta\Sigma$ -PWM ist ein dreiphasiger P Stromregler [10]. In Kombination mit der hohen Kleinsignalbandbreite lassen sich große Proportionalitätsfaktoren einstellen, sodass die stationäre Abweichung minimiert wird [10]. In Abb. 5.1 ist die $\Delta\Sigma$ -PWM mit dem proportionalen Stromregler in orange abgebildet. Um die stationäre Genauigkeit zu erhöhen, wird die Elektromagnetische Kraft (EMK) \underline{u}_h vorgesteuert. Der Vorteil eines Reglers ohne Integralanteil ist, dass keine zusätzliche Begrenzung eingeführt werden muss und somit die phasenrichtige Modulation bei Übermodulation der $\Delta\Sigma$ -PWM sichergestellt ist. Der Verstärkungsfaktor ergibt sich aus dem Produkt der Normierung für Strom und Spannung durch die $\Delta\Sigma$ -M, sowie dem eingestellten Verstärkungsfaktor P .

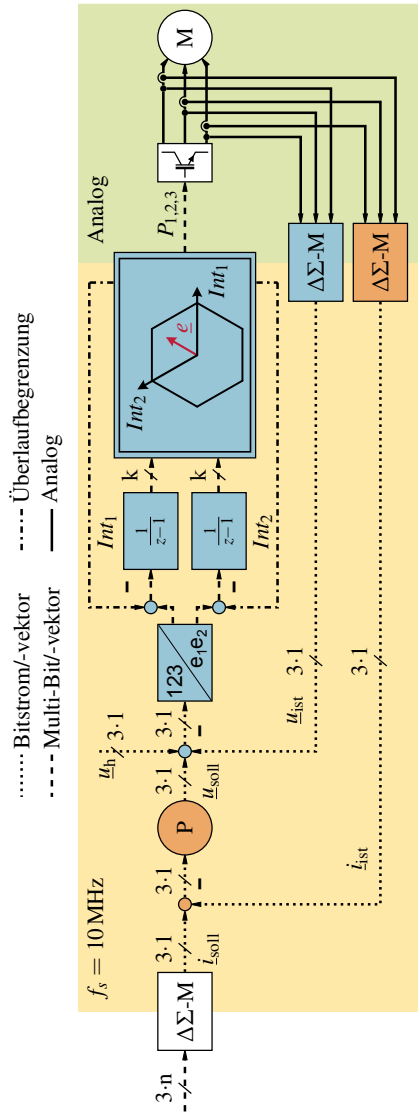


Abbildung 5.1: $\Delta\Sigma$ -PWM mit interner **Spannungsregelung** und **P Stromregelung** [10].

5.1.2 Proportional Integral Stromregler im 120° System

Für stationäre Genauigkeit im Stromregelkreis wird ein Regler mit Integralanteil benötigt. Ein Ansatz ist, die PI Regler nach der Transformation in das 120° Fehler System einzusetzen, um für die zwei Freiheitsgrade eines dreiphasigen Systems stationär genau zu regeln. Als Begrenzung für den Integralanteil wird $\frac{2}{\sqrt{3}}$ der normierten Zwischenkreisspannung verwendet, da dieser Wert der maximalen Phasenspannung bei sinusförmiger Ansteuerung ohne Übermodulation im 120° System entspricht

$$\max \left(\left| \cos(\varphi) - \frac{\sin(\varphi)}{\sqrt{3}} \right| \right) = \max \left(\left| \frac{2 \sin(\varphi)}{\sqrt{3}} \right| \right) = \frac{2}{\sqrt{3}}. \quad (5.1)$$

Für die Begrenzung muss beachtet werden, dass keine Übersteuerung des digitalen $\Delta\Sigma$ -M der $\Delta\Sigma$ SV Operation auftritt. Die daraus resultierende Regelung ist in Abb. 5.2 dargestellt.

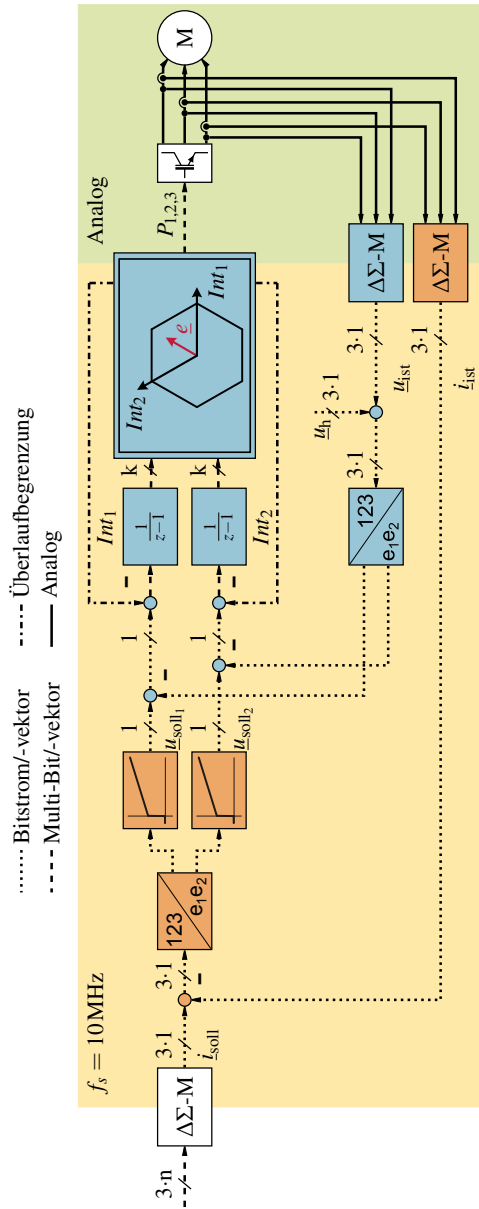


Abbildung 5.2: $\Delta\Sigma$ -PWM mit interner Spannungsregelung und PI Stromregelung im 120° System.

5.1.3 Proportional Integral Stromregler im d/q System

In den meisten feldorientierten Regelungen ist eine Sollwertvorgabe und Regelung im d/q System gebräuchlich. Hierbei werden die PI Regler für die d Achse und die q Achse auf die Zwischenkreisspannung begrenzt. Die Regelungsstruktur ist in Abb. 5.3 dargestellt. Der Nachteil der Stromregelung im d/q System ist, dass der elektrische Winkel der Maschine für die Koordinatentransformation bekannt sein muss. Neben der Verwendung von Winkelgebern gibt es mehrere Ansätze für drehgeberlose Winkelerfassung. In Abschnitt 5.3 auf Seite 96 wird die Implementierung nach [KTS18a] des Vorschlags aus [10] für die $\Delta\Sigma$ -PWM diskutiert.

5.1.4 Versuch

Zur Evaluierung der erzielten Bandbreite wurden die Regelungsalgorithmen auf dem in Abschnitt 4.1 vorstellten Prüfstand implementiert und vermessen. Unter der Annahme einer quasi Laufzeit freien Regelung kann die Verstärkung (P) der proportionalen Regler beliebig eingestellt werden und skaliert die Polstelle der Strecke im geschlossenen Regelkreis. Aufgrund der im Vergleich zur dominierenden Statorzeitkonstanten geringen, jedoch vorhandenen Verzögerung der Spannungsregelung, der ADU und der $\Delta\Sigma$ SV, existiert für P eine obere Schranke. In der praktischen Erprobung hat sich ein normiertes P von Neun als sinnvoll herausgestellt. Normiert ist P hinsichtlich des maximal messbaren Stromes und der Spannung, da die Messgrößen in Bitströmen repräsentiert sind. Aufgrund der gleichen Skalierungen zwischen Nenngröße und maximal messbarer Größe bei Strom und Spannung entspricht die Normierung dem Verhältnis zwischen Nennspannung und Nennstrom. Um den resultierenden Sollwert in einem Bitstrom repräsentieren zu können, sind mehrere Lösungsansätze möglich. Ohne die Erweiterung auf Multibit-Bitströme kann die Spannungsmessung mit einer $\Delta\Sigma$ SV Operation mit $\frac{1}{P}$ skaliert werden, was effektiv eine Verstärkung der Sollwerte darstellt. Aufgrund der internen Datenverarbeitung des $\Delta\Sigma$ -PWM ist es möglich, die Spannungssollwerte nach einer Bipolaren Interpretation mit P zu skalieren. Die nachfolgende Datenverarbeitung enthält keine Konvertierung in einen Bitstrom. Die beiden Lösungen unterscheiden sich durch die mögliche Schrittweite des Regelfehlers und somit der Wahl von N_1 für eine bestimmte Schaltfrequenz. Die Skalierung hat inhärente Nachteile, wie aus Abb. 3.15 auf Seite 22 ableitbar [Kurve: Analog mit $\Delta\Sigma$ -Modulator] und in [HKKS16] dargestellt. Für die Sollgröße Eins, welche im Skalierten Bitstrom $\frac{1}{P}$ entspricht, nimmt die Signalqualität bei $P = 9$ um etwa 20 dB ab. Eine bessere Lösung stellt somit ein digitaler $\Delta\Sigma$ -M mit einem mehrstufigen Quantisierer [34] dar. Somit entsteht erhöhter Verbindungsaufwand zwischen den Reglern, den Koordinatentransformationen und dem $\Delta\Sigma$ -PWM. Der Verbindungsaufwand ist geringer, als bei einer klassischen Multibit-Regelung. Zum Beispiel um die Quantisierungsstufen -9 bis $+9$ darstellen zu können, werden 5 bit benö-

tigt. In der Implementierung wurde der Ansatz mit Multibit-Bitströmen umgesetzt.

Die PI Regler im 120° System und im d/q System haben als weiteren Freiheitsgrad die Integratorzeitkonstante. Mit dieser kann die langsamste Zeitkonstante der Strecke kompensiert werden. Im gegebenen Fall ist dies die Statorzeitkonstante. Die geringen Verzögerungen durch die zusätzlichen Datenverarbeitungsschritte der Park-Transformation und der inversen Park-Transformation in $\Delta\Sigma\text{SV}$ bei den PI Reglern im d/q System werden bei der Auslegung nicht berücksichtigt.

Als Testsignal wird im festgebremsten Zustand der Maschine eine Sinusschwingung mit veränderlicher Frequenz verwendet. Im Falle der proportionalen Regler und der PI Regler im 120° System wird ein dreiphasig rotierender Strom vorgegeben und mit den Strangströmen verglichen. Im Fall der PI Regler im d/q System wird eine Sinusschwingung auf dem d Strom vorgegeben und mit den transformierten Strangströmen im d/q System ausgewertet.

Die $\Delta\Sigma\text{-PWM}$ wird mit dem in Abschnitt 4.4 vorgestellten Schaltfrequenzregler auf eine mittlere Schaltfrequenz von 16 kHz geregelt.

5.1.5 Ergebnisse

Aus den Bodediagrammen in Abb. 5.4 können die dynamischen Eigenschaften der Stromregler abgelesen werden. Die P Regler im dreiphasigen System und die PI Regler im 120° System weisen eine Bandbreite von 7 kHz auf. Bei Frequenzen bis 700 Hz ist der Einfluss der Integralanteile sichtbar. Sie sorgen für eine bessere stationäre Genauigkeit. Bei höheren Frequenzen dominiert der Einfluss der proportionalen Regler, welche bei beiden Implementierungen gleich sind. Somit ist die Beobachtung des gleichen Verlaufs der Bodediagramme plausibel. Die PI Regler im d/q System weisen eine Bandbreite von 8 kHz auf. Bedingt durch die zusätzlichen $\Delta\Sigma\text{SV}$ Operationen der Transformationen werden in diesem Fall jeweils in der Vorwärtsübertragungsfunktion, sowie in der Rückführung eine zusätzliche Polstelle eingeführt. Die zusätzliche, nicht modellierte, Verzögerung sorgt für eine Überkompensation des Regelfehlers und somit eine Verstärkung

im geschlossenen Regelkreis. Zusätzlich fügen die $\Delta\Sigma$ SV Operationen der Transformationen zusätzliches Rauschen ins Signal ein.

Im Vergleich zu klassischer Motorregelung mit Festfrequenz-PWM ist die resultierende Bandbreite von $\leq 7\text{--}8\text{ kHz}$ im Stromregler bei einer mittleren Schaltfrequenz von 16 kHz dynamisch. Klassische Regelungen erreichen bei dieser Schaltfrequenz Bandbreiten zwischen $2,20\text{ kHz}$ und 8 kHz [8]. Somit ist die hohe Dynamik der $\Delta\Sigma$ -PWM verifiziert. Die Beobachtung wird durch die Ergebnisse in [10] untermauert, wo mit proportionalen Reglern bei einer mittleren Schaltfrequenz um 10 kHz eine Bandbreite von 5 kHz erreicht wird.

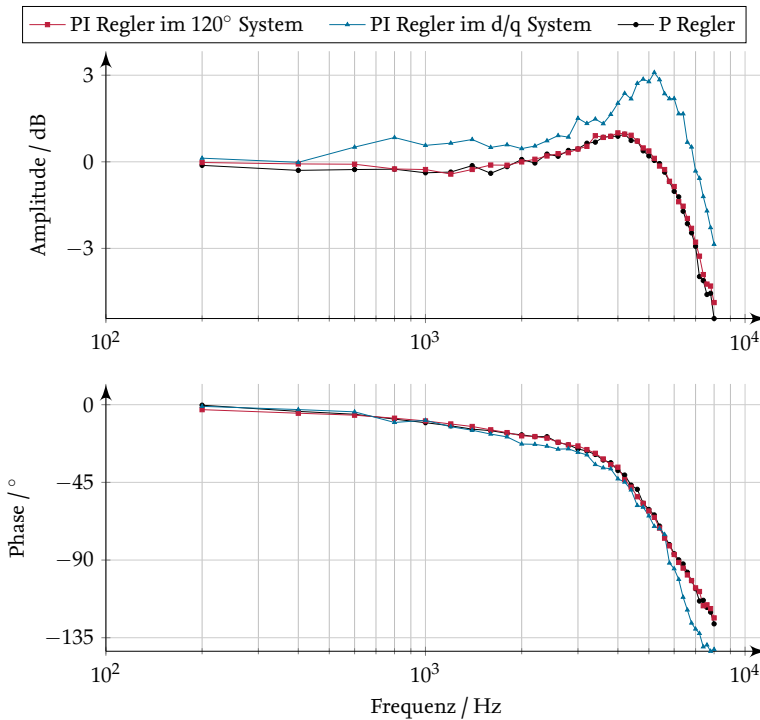


Abbildung 5.4: Bodediagramme der Stromregelungen.

Die erreichten Signalqualitäten bei einer Schwingung mit 100 Hz und unterschiedlichen Aussteuerungen sind in Abb. 5.5 dargestellt. Bedingt durch den hohen Verstärkungsfaktor treten Oberschwingungen auf. Die Signal-

qualität der proportionalen Regler im dreiphasigen System und die PI Regler im 120° System weisen die gleiche Signalqualität zwischen 34 dB und 38 dB auf. Die PI Regler im d/q System erreichen eine geringere Signalqualität zwischen 28 dB und 30 dB, wobei die Signalqualität mit steigenden Amplituden tendenziell zunimmt. Die Signalqualität unterstützt die Erkenntnis, dass die Verzögerung der $\Delta\Sigma$ SV Operationen in der Auslegung berücksichtigt werden sollte. Die zusätzlichen Verzögerungen bewirken, dass der geschlossene Kreis eine geringere Dämpfung aufweist, was zu stärkerem Überschwingen führt. Zusätzlich fügen die $\Delta\Sigma$ SV basierten Transformationen im Regelkreis (Park-Transformation und inverse Park-Transformation) Rauschen ein. Bei einer Einheitsskalierung mit $\Delta\Sigma$ -M nach [25] sinkt die Signalqualität um etwa 3 dB pro Operation, wie in Abschnitt 3.3 dargelegt. Somit ist eine um 6 dB schlechtere Performance zu erwarten. Des Weiteren sind die Skalierungsfaktoren ungleich Eins und zeitlich variabel, was eine weitere Reduktion der Signalqualität bewirkt. Folglich ist die Signalqualitätsreduktion um 6 dB bis 8 dB plausibel.

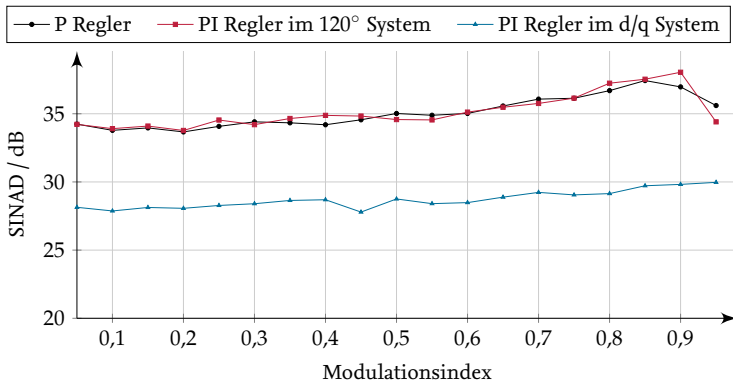


Abbildung 5.5: Signalqualität der Stromregelung über dem Aussteuerungsgrad.

Der PI Regler im d/q System bietet mehrere Vorteile im Vergleich zu den anderen beiden vorgestellten Regelverfahren. Mit der d- und der q-Komponente lässt sich der Feld erzeugende Strom unabhängig vom Drehmoment erzeugenden Strom vorgeben und zusätzlich sind die Sollwerte der Grundschwingung Gleichgrößen. Das heißt, die gesamte Dynamik des

Reglers kann zur Störausregelung verwendet werden und die stationäre Genauigkeit des PI Reglers kommt bei der Grundschwungung zur Wirkung. Dieses Prinzip wird im nachfolgenden Abschnitt für die harmonische Regelung aufgegriffen.

5.2 Harmonische Regelung

Bei harmonischen Prozessen mit Oberschwingungen, wie sie in der Antriebsregelung zum Beispiel bei der Stromregelung von Maschinen mit einer nicht sinusförmigen EMK oder der Drehgeberauswertung vorkommen, können klassische PI-Regler keinen stationären Zustand erreichen. Dies liegt daran, dass die Verstärkung des Integralanteils nur für Gleichgrößen gegen Unendlich geht. Für Wechselgrößen nimmt die Verstärkung mit steigender Frequenz ab. Für Systeme mit einer dominanten Grundschwungung wird zum Beispiel die Park-Transformation verwendet, um die Grundschwungung mittels Frequenzverschiebung in eine stationäre Größe abzubilden. Die Oberschwingungen bleiben alternierende Signale mit den damit verbundenen Problemen für einen PI-Regler. Ein resonanter Regler erreicht im Gegensatz dazu stationäre Genauigkeit bei einer festgelegten Frequenz, siehe Abb. 5.6 [15]. Die nachfolgende Diskussion basiert auf [MTK18].

Ein resonanter Regler kann als Grundelement parallel mit weiteren resonanten Reglern geschaltet werden, sodass sich ein resonanter Regler ergibt, der bei mehreren Frequenzen stationär genau ist. In der Regel werden die Eigenfrequenzen der Grundelemente so ausgelegt, dass eine größere Anzahl von Oberschwingungen der Grundfrequenz stationär ausgeregelt wird.

5.2.1 Resonante Regelungsstrukturen

Ein resonanter Regler erreicht die stationäre Genauigkeit bei einer Frequenz durch drei Schritte. Die Modulation der Regelabweichung mit der erwarteten Schwingung, was zu der Amplitude der Schwingung führt, die Integration der Amplitude und die Demodulation um die Amplitude in die entsprechende Schwingung zu transformieren. Für den Fall, dass zwischen dem Referenzsignal und dem geregelten Signal eine Phasenverschiebung

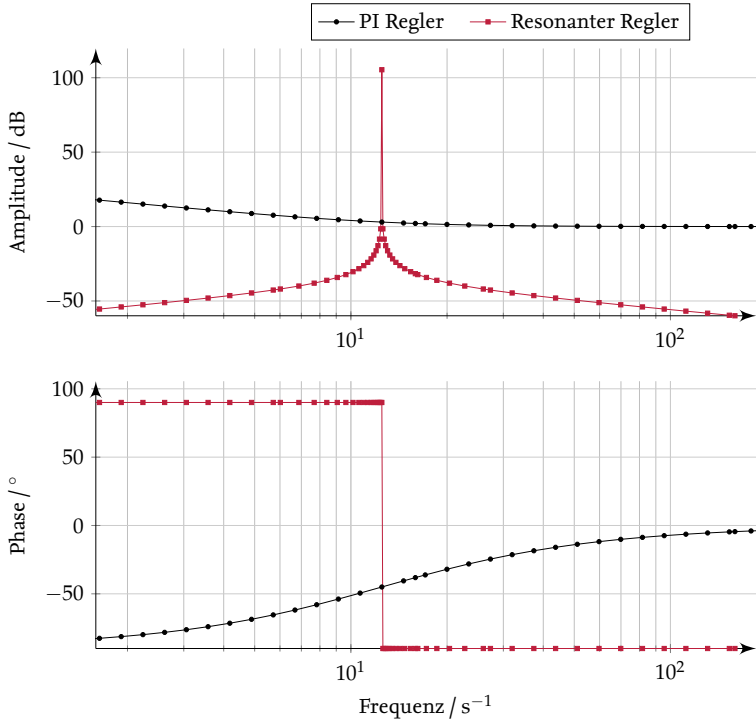


Abbildung 5.6: Bodediagramm eines PI-Reglers und eines resonanten Reglers [MTK18].

auftritt ist die Struktur aus [15] mit paralleler Sinus und Cosinus Modulation empfehlenswert. Folglich ergibt sich für einen parallelen proportionalen Regler, den grundschwingungs Regler und überschwingungs Regler basierend auf dem Regelfehler und einem Referenzwinkel, der mit der Frequenz der Grundschwingung rotiert die in Abb. 5.7 dargestellte Regelung.

Wie bereits in [15] dargestellt, ist der resonante Regler, bis zur Shannon-Grenze stabil auslegbar. Für Überschwingungen höherer Ordnung kann diese Bedingung bei klassisch abgetasteten Systemen ($f_s \leq 40 \text{ kHz}$) zu Einschränkungen führen. Eine Lösung ist die Verwendung von $\Delta\Sigma\text{SV}$, die mit einer Abtastrate von $\geq 10 \text{ MHz}$ Signale bis 5 MHz verarbeiten kann.

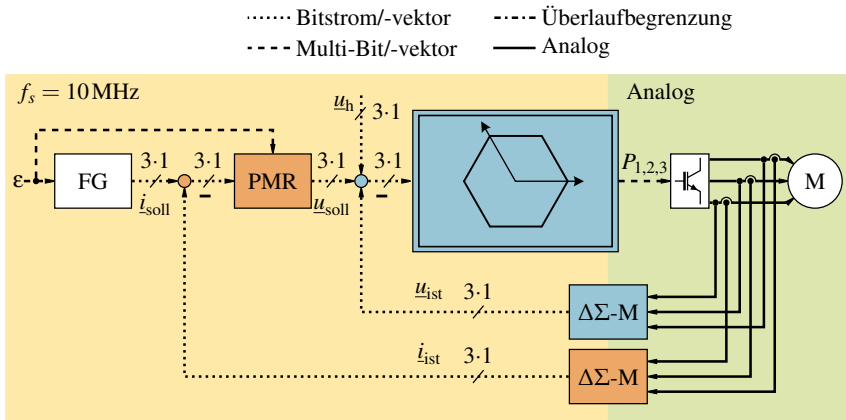


Abbildung 5.8: Regelkreis mit Führungsgrößengenerator (FG) zur Evaluation der resonanten Regler (PMR) in $\Delta\Sigma\text{SV}$ [MTK18].

Somit eignet sich die $\Delta\Sigma$ SV um resonante Regler zu Implementieren. Bei gängigen Anwendungen, wie zum Beispiel der Regelung von Transversalflussmaschinen [15], oder auch in Beobachterstrukturen zur Geberfehlerkorrektur [1]. Der Vorteil der $\Delta\Sigma$ SV liegt im Implementierungsaufwand, da sie mit unter 72 % der Logikgatter der Implementierung in DSV auskommt, wie aus Tabelle 5.1 ersichtlich.

Tabelle 5.1: Ressourcenverbrauch der Operation Begrenzung auf einem FPGA [MTK18].

Implementierung	Logikgatter
DSV	5693
$\Delta\Sigma V$	4097

Die effizientere Implementierung in Δ SSV ermöglicht mit genügend schnellen ADU einen Regeltakt von 20 % des FPGA Basistaktes, die Implementierung in DSV erreicht maximal 14 %. Das bedeutet, dass mit der gleichen Recheneinheit das Regelgesetz etwa anderthalb mal so oft berechnet werden kann.

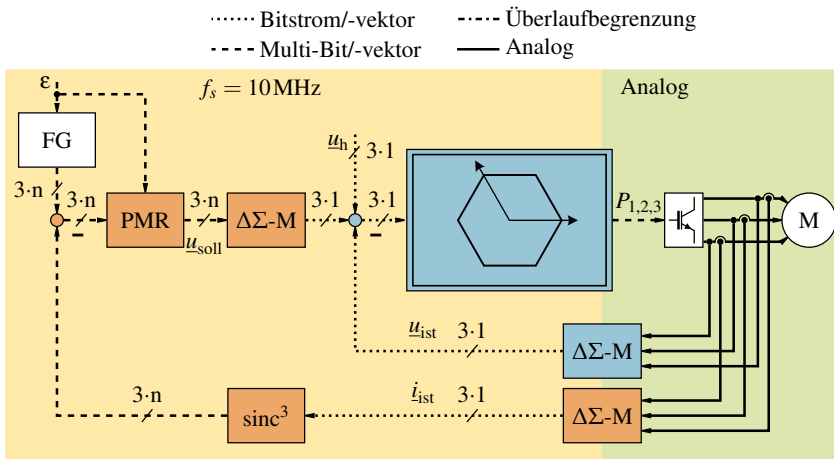


Abbildung 5.9: Regelkreis mit Führungsgrößengenerator (FG) zur Evaluation der resonanten Regler (PMR) in DSV [MTK18].

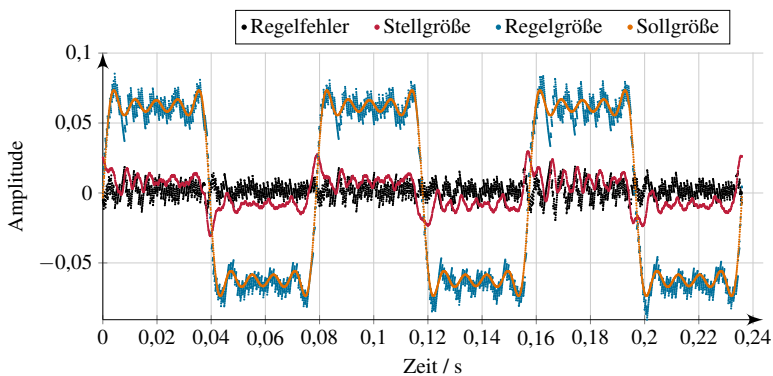


Abbildung 5.10: Oberschwingungsbehaftete Stromregelung mit DSV [MTK18].

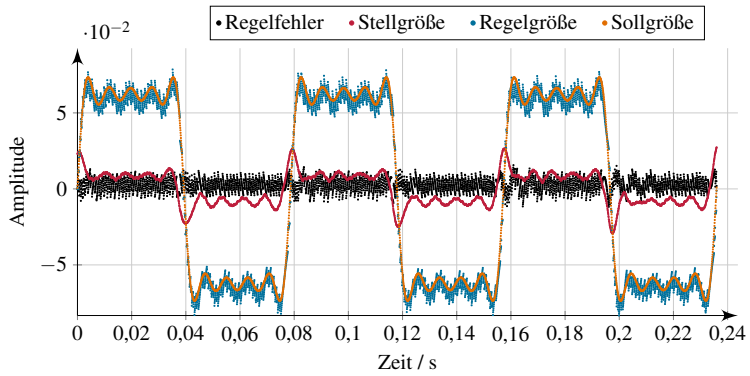


Abbildung 5.11: Oberschwingungsbehaftete Stromregelung mit $\Delta\Sigma\text{SV}$ [MTK18].

5.3 Drehgeberlose Geschwindigkeitsregelung

Bei der Implementierung mit drei proportionalen Phasenstromreglern weist der eingeschwungene Zustand eine Regelabweichung auf. Normalerweise ist dieses Verhalten unerwünscht. In [10] wird die These aufgestellt, dass der Regelfehler proportional zur EMK ist. Daraus folgt, dass eine drehgeberlose Geschwindigkeitsregelung basierend auf diesem Signal realisiert werden kann. In [KTS18a], auf der dieser Abschnitt basiert, ist die These analysiert und experimentell verifiziert.

Der Regelfehler wird durch einen Verstärkungsfaktor größer 1 in den Stromreglern sowie die Vorsteuerung der geschwindigkeitsproportionalen EMK reduziert. Experimentell wurde eine Abweichung von $\leq 4\%$ bei einer sinusförmigen Anregung der Phasenströme identifiziert. Die tiefpassgefilterte Messung ist in Abb. 5.13 auf der nächsten Seite dargestellt. In [KTS18a] wurde neben dem Stromregelfehler auch der Spannungssollwert als Eingangssignal für die drehgeberlose Winkelschätzung untersucht. Dieses Signal entspricht dem verstärkten Stromregelfehler mit zusätzlicher Aufschaltung der EMK. Wie in Abb. 5.14 dargestellt, wirkt die EMK in der q-Achse und die Phasenregelschleife (PLL) zur Winkelschätzung verwendet die d-Achse. Somit sind diese Signalanteile voneinander entkoppelt. Die Verstärkung und die Addition der EMK-Vorsteuerung in $\Delta\Sigma\text{SV}$ führt zu einem schlechteren SINAD. Die tiefpassgefilterten Sollspannungen für den Arbeitspunkt sind in Abb. 5.12 dargestellt. Aufgrund der schlechteren Signalqualität der Sollspannungen wird im Folgenden die Winkelschätzung basierend auf dem Stromregelfehler vorgestellt.

5.3.1 Schätzung der elektromotorischen Kraft

Die vereinfachte regelungstechnische Darstellung der $\Delta\Sigma\text{-PWM}$ aus [10] ist in Abb. 5.14 mit einer PLL zur Winkelschätzung abgebildet. Die Dynamik des Spannungsregelkreises wird, wie identifiziert, als Laufzeit modelliert und die Stromeinprägung in Abhängigkeit von der angelegten Spannung als Verzögerung erster Ordnung. Die induzierte EMK wird als Störgröße

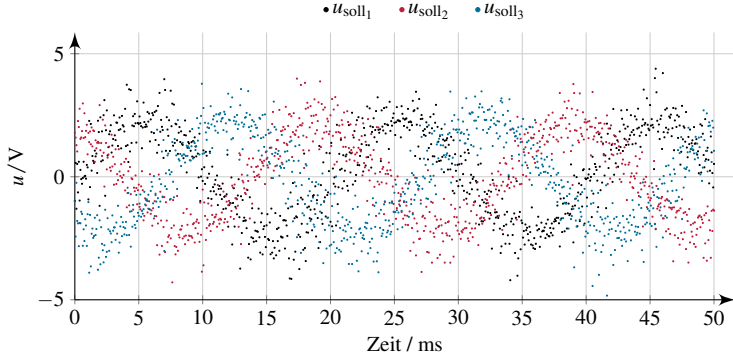


Abbildung 5.12: Gemessene Sollspannungen bei proportionaler Phasenstromregelung mit EMK Vorsteuerung [KTS18a].

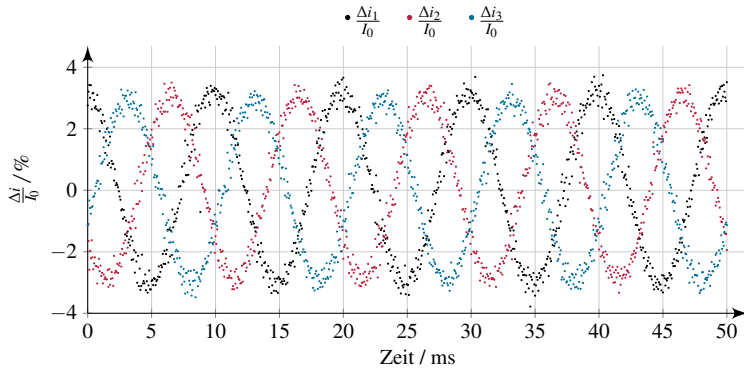


Abbildung 5.13: Gemessener relativer Stromregelfehler bei proportionaler Phasenstromregelung mit EMK Vorsteuerung [KTS18a].

\underline{u}_h modelliert. Zusätzlich ist die verwendete PLL eingezeichnet. Die d-Achse des Stromregelfehlers dient als Eingangssignal. Mit der geschätzten Geschwindigkeit wird die EMK Vorsteuerung berechnet und der geschätzte Winkel wird für die (inversen) Park Transformationen verwendet. Somit ist es möglich, die Sollströme im dq-System vorzugeben. Die EMK berechnet sich aus der Drehzahl und der EMK-Konstanten nach

$$\hat{u}_h = \omega \cdot k_{\text{EMK}}. \quad (5.2)$$

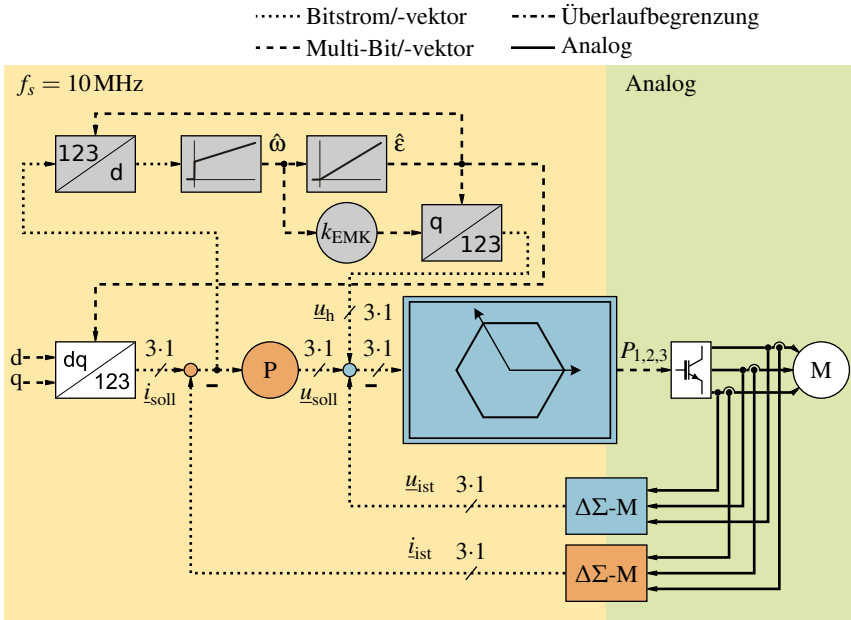


Abbildung 5.14: Drehwinkelschätzung für die $\Delta\Sigma$ -PWM basierend auf dem Regelfehler der **proportionalen Stromregler** mittels PLL [KTS18a].

Aufgrund der Eigenschaft, dass der Winkel von einer Überlaufvariable mit Überschlagn im Bereich $[-\pi, \pi]$ repräsentiert wird und anschließend als Argument der nichtlinearen Funktionen Sinus und Cosinus verwendet wird, ist die Repräsentation als Multibit-Variable der Repräsentation als Bit-Strom vorzuziehen.

Die Parameter des PI-Filters der PLL werden am linearen Modell bestimmt. Dazu wird die Koordinatentransformation durch die um k_{CT} verstärkte Differenz des geschätzten Winkels und des transformierten Eingangssignals ersetzt. Somit ergibt sich ein lineares System zweiter Ordnung für das Übertragungsverhalten der PLL. Für eine gewünschte Bandbreite

ω_n und eine vorgegebene Dämpfung D können die Parameter berechnet werden [37]

$$V_P = \frac{2 \cdot D \cdot \omega_n}{k_{CT}} \quad (5.3a)$$

$$T_i = \frac{2 \cdot D}{\omega_n}. \quad (5.3b)$$

Die Eingangsbitströme setzen sich aus den Nutzdaten und hochfrequentem Rauschen zusammen, daher ist die Bandbreite der PLL für maximal schnelles Einschwingen heuristisch bestimmt, sodass das Rauschen hinreichend gedämpft wird. Die Dämpfung wird auf 0,70 gesetzt.

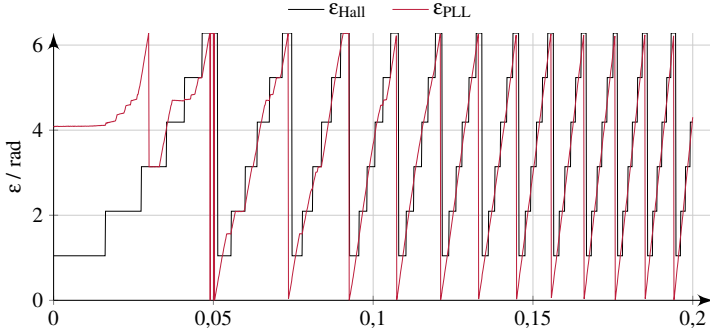
5.3.2 Regelkreis

Für die Geschwindigkeitsregelung wird die Reglerkaskade um einen PI-Drehzahlregler in $\Delta\Sigma SV$ erweitert. Um das hochfrequente Rauschen der Geschwindigkeit aus der PLL (ω_{pll}) zu dämpfen wird ein schnelles sinc^3 -Filter verwendet, das eine kleine Gruppenlaufzeit im Vergleich zur Zeitkonstante des Motors hat, damit keine signifikante Auswirkung auf die Kleinsignalbandbreite des äußeren Regelkreises entsteht. Der Regelkreis ist in Abb. 5.15 dargestellt.

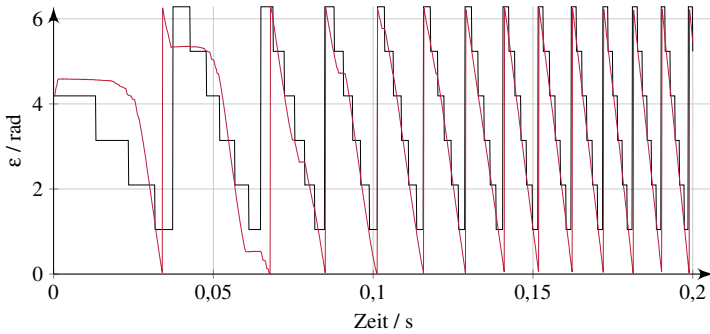
5.3.3 Ergebnisse

Um die Performance der Winkelschätzung zu evaluieren, wird die Maschine aus dem Stillstand mit einem q-Strom Sprung angeregt. Bei Stillstand bis zum Einrasten der PLL wird der Winkel mit einem Hall-Sensor gemessen. Nach dem Einrasten der PLL wird auf Sinuskommutierung basierend auf dem geschätzten Winkel umgeschaltet.

In Abb. 5.16 ist der geschätzte Winkel und der per Hall-Sensor gemessene für einen positiven und negativen q-Strom Sprung dargestellt. Wie aus den Messungen ersichtlich wird, ist die PLL in unter 0,08 s eingerastet und zeigt eine Winkelabweichung von $0,10^\circ$ für positive und $0,30^\circ$ für negative Drehrichtung. Die unterschiedliche Winkelabweichung von positiver zu negativer Drehrichtung ist auf eine ungenaue Positionierung des Hall-



(a) Positiver q-Strom Sprung aus dem Stillstand.



(b) Negativer q-Strom Sprung aus dem Stillstand.

Abbildung 5.16: Gemessener Winkel (Hall-Sensor) und mittels PLL beobachteter Winkel [KTS18a].

Sensors zurückzuführen. Dieses Ergebnis belegt die Dynamik und Funktion des Winkelbeobachters.

Basierend auf der PLL kann eine Drehzahlregelung aufgebaut werden.

In Abb. 5.17 sind ein negativer und ein positiver Sprung auf der Soll-drehzahl gezeigt. ω_{PLL} folgt der gemessenen Geschwindigkeit mit etwas Rauschen. Das Rauschen in allen Punkten, und besonders in den Bereichen von 0,50 s bis 0,70 s und von 1,50 s bis 1,70 s, auf ω_{PLL} resultiert aus der dynamischen Auslegung. Bei einer Geschwindigkeitsregelung ist der Hauptfokus auf der Geschwindigkeit, die nahezu konstant und ohne Abweichung ist.

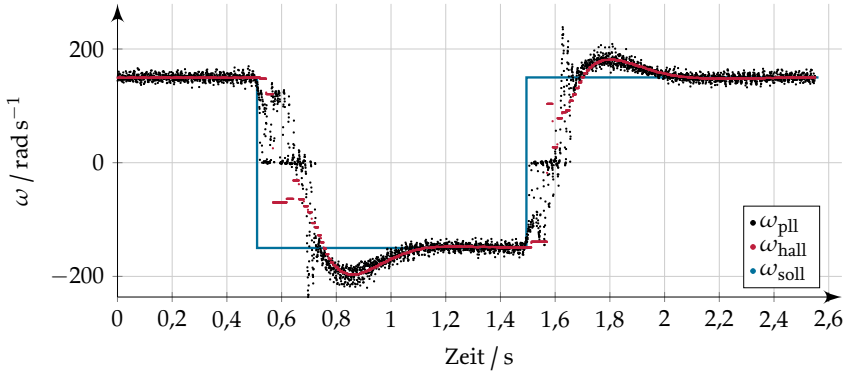


Abbildung 5.17: Drehgeberlose Geschwindigkeitsregelung (gemittelt über 10 Datenpunkte) [KTS18a].

Zusammenfassend wurde mit den Messungen das Konzept der Winkelschätzung aus dem Stromregelfehler in [10] verifiziert. Bei der gewählten Auslegung lassen sich Einrastzeiten von 80 ms und Abweichungen um $0,20^\circ$ erzielen. Mit dieser Winkelschätzung lässt sich eine Geschwindigkeitsregelung aufbauen, die eine hohe Dynamik aufweist und stationär genau ist.

6 Zusammenfassung und Ausblick

Im Rahmen dieser Arbeit wurden lineare und nichtlineare $\Delta\Sigma$ SV Operationsklassen entwickelt und vorgestellt. Um die Leistungsfähigkeit zu vergleichen, wurden aus der Literatur bekannte Verfahren zum Teil weiterentwickelt und die dynamischen Eigenschaften verglichen. Besonderes Augenmerk wurde auf die Gruppenlaufzeit und die erzielbare Signalqualität (SINAD) gelegt. Zusätzlich wurde der Implementierungsaufwand in Logikgattern betrachtet. Dabei stellt sich die neu entwickelte Operationsklasse Bipolare Interpretation für lineare Operationen als beste Lösung dar. Sie erreicht das beste SINAD bei der gleichen Gruppenlaufzeit wie die anderen Operationsklassen basierend auf Modulatoren zweiter Ordnung. Des Weiteren lässt sich die Gruppenlaufzeit durch die Parametrierung des $\Delta\Sigma$ -M beeinflussen. Für eine Operation mit einem $\Delta\Sigma$ -M mit verzögerungsfreien Parametersatz sinkt die Signalqualität bei hohen Aussteuerungen auf die Signalqualität einer Operation mit einem $\Delta\Sigma$ -M erster Ordnung ab. Aufgrund der einfachen Abbildungsvorschrift als vorbereitenden Schritt, lässt sich die Operationsklasse effizient implementieren.

Für nichtlineare Operationen lässt sich die gefundene Operationsklasse basierend auf schnellen Tiefpassfiltern für eine generelle Implementierung nichtlinearer Funktionen verwenden. Wie in den Beispielen gezeigt, kann es für einzelne Operationen effizientere spezialisierte Implementierungen geben. Diese erfüllen jedoch nicht die Forderung, ein allgemeingültiger Ansatz zur Darstellung von nichtlinearen Funktionen zu sein.

Weiter wurde untersucht, was mit den repräsentierten Werten passiert, wenn Logik Operationen auf Bitströme angewendet werden. Bei der Betrachtung der Gruppenlaufzeit, Null bis ein Takt, und des Implementierungsaufwands, ein Logikgatter, weisen diese Lösungen sehr gute Resultate auf. Aufgrund der Anforderung von statistischer Unabhängigkeit an die Eingangsbitströme bei den Logikoperationen mit zwei Eingängen, ist

für den realen Einsatz einzig die NOT Operation als Negation unter dem Aspekt der Signalqualität sinnvoll.

Mit der Operationsklasse Bipolare Interpretation und den nichtlinearen Operationen basierend auf schnellen Filtern lassen sich komplexe regelungstechnische Strukturen mit geringer Gruppenlaufzeit und guter Signalqualität abbilden. Um diese hochfrequenten Signale mit Leistungselektronik an einer dreiphasigen Last zur Wirkung zu bringen, wurde der $\Delta\Sigma$ -PWM nach [10] verwendet.

Zunächst wurden die unterschiedlichen Operationsmodi des $\Delta\Sigma$ -PWM untersucht. Dabei ist aufgefallen, dass die Operationsmodi Auslassen von Nullvektoren und Übermodulation nicht nur durch den $\Delta\Sigma$ -PWM, sondern auch durch die Totzeitüberwachung erzeugt werden, was zur Folge hat, dass der Modulator intern entweder im Betriebsmodus Raumzeigermodulation, oder Übermodulation mit Überlaufbegrenzung operiert. Die anderen Operationsmodi werden bereits durch die Totzeitüberwachung erzeugt.

Eine weitere Herausforderung der $\Delta\Sigma$ -PWM ist die variierende Schaltfrequenz. Um diese zu stabilisieren wurde ein Schaltfrequenzregler vorgestellt. Dieser regelt bei Raumzeigermodulation in jedem PWM Halbzyklus die Schaltfrequenz und in allen anderen Operationsmodi im PWM Vollzyklus. Dies bringt den Vorteil, dass die Schaltfrequenz mit der maximalen Dynamik geregelt werden kann und die Schaltzyklus Degeneration durch Auslassen von einzelnen oder allen Nullvektoren keinen Einfluss hat.

Die Kombination der $\Delta\Sigma$ SV mit dem $\Delta\Sigma$ -PWM ergibt eine Lösung zur Regelung von dreiphasigen Lasten, wie zum Beispiel Drehstrommotoren. In dieser Arbeit wurden speziell permanenterregte Synchronmotoren untersucht.

Die Grundlage der Antriebsregelung stellt die Stromregelung dar. Diese wurde anhand unterschiedlicher Ansätze implementiert. Die klassische Implementierung stellt die dreiphasige Stromregelung mit proportionalen Reglern dar. Aufgrund der hohen Kleinsignalbandbreite der durch den $\Delta\Sigma$ -PWM unterlagerten Spannungsregelung lassen sich Verstärkungsfaktoren verwenden, die eine geringe stationäre Abweichung erreichen. Für stationäre Genauigkeit empfiehlt es sich jedoch PI Regler einzusetzen. Diese

wurden für zwei Implementierungen verglichen. Zum einen im 120° System, zum anderen im d/q-System. Dabei zeigte sich, dass bei gleichen Parametern die Phasenstromregler und im 120° System die gleiche Bandbreite von 7 kHz und im d/q System 8 kHz erreichen. Der Regler im d/q-System regelt für die Grundschiwingung eine Gleichgröße, was die stationäre Genauigkeit der Grundschiwingung verbessert. Für die PI Regler im 120° System und die Phasenstromregler ist die Grundschiwingung eine Wechselgröße, die somit innerhalb der Reglerbandbreite liegen muss. Zusammenfassend ist die Regelung im d/q-System zu bevorzugen.

Eine weitere Anwendung ist die Regelung periodischer Prozesse oder die Drehmomentregelung von Transversalflussmaschinen. Diese haben häufig einen hohen Oberschwingungsanteil, der mit resonanten Reglern stationär genau ausgeregelt werden kann. Diese wurden in $\Delta\Sigma\text{SV}$ implementiert und mit einer klassischen Implementierung in DSV verglichen. Dabei zeigte sich die $\Delta\Sigma\text{SV}$ mindestens gleichwertig bei der Regelqualität, jedoch effizienter in der Implementierung.

Abschließend wurde eine Drehgeberlose Positionsbestimmung in der Anwendung einer Drehzahlregelung vorgestellt. Diese weist eine hohe Dynamik auf und regelt die Geschwindigkeit stationär genau.

In zukünftigen Untersuchungen können die entwickelten Operationsklassen verwendet werden, um eine direkte Selbstregelung von elektrischen Maschinen zu implementieren. Dieser Ansatz verspricht gute Regeleigenschaften, da die Abtastfrequenz der größte Einflussfaktor bei der direkten Selbstregelung auf die erzielbare Regelqualität darstellt.

Literatur

- [1] Albrecht, C., J. Klöck, O. Martens und W. Schumacher: Online Estimation and Correction of Systematic Encoder Line Errors. Machines, 5(1), 2017.
- [2] Ardalan, S. H. und J. J. Paulos: An analysis of nonlinear behavior in delta - sigma modulators. Circuits and Systems, IEEE Transactions on, 34(6):593–603, 1987.
- [3] Bernoulli, J. und C. Huygens: Wahrscheinlichkeitsrechnung (Ars conjectandi), Bd. 2. W. Engelmann, 1899.
- [4] Bradshaw, J., U. Madawala, N. Patel und M. Vilathgamuwa: A Bit-Stream based space vector modulator. In: International Power Electronics Conference (IPEC), S. 855–861, 2010.
- [5] Bradshaw, J. B.: Bit-Stream Control of Doubly Fed Induction Generators. Doktorarbeit, University of Auckland, 2012.
- [6] Dwersteg, B.: QMot QBL4208 family. Trinamic, 11. 2010.
- [7] Gonzalez, E. P. und J. D. Reiss: Idle tone behavior in sigma delta modulation. In: Audio Engineering Society 122nd Convention Papers CD-ROM (2007 May), convention paper, Bd. 7108, 2007.
- [8] Gröling, C.: Optimierungspotenzial bei Servoumrichtern für permanenterregte Synchronmaschinen. Doktorarbeit, TU Braunschweig, 2009.
- [9] Holmes, D. G. und T. A. Lipo: Pulse width modulation for power converters: Principles and practice, Bd. 2 d. Reihe IEEE Press series on power engineering. IEEE Press, Piscataway, NJ, 2003.
- [10] Homann, M.: Hochdynamische Strom- und Spannungsregelung von permanenterregten Synchronmaschinen auf Basis von Delta-Sigma Bitströmen. Doktorarbeit, TU Braunschweig, Braunschweig, 2016.
- [11] Homann, M. und W. Schumacher: Stromrichter und Computerprogramm. Patent, 06. 2014. DE 102014108667 A1.
- [12] Homann, M. und W. Schumacher: High Bandwidth Phase Voltage and Phase Current Control Loop of a Permanent Magnet Synchronous Motor based on Delta Sigma Bitstreams. In: 18th European Conference on Power Electronics and Applications (EPE), 2016.
- [13] Kershaw, S. M. und M. B. Sandler: Digital signal processing on a sigma-delta bitstream. In: Oversampling Techniques and Sigma-Delta Modulation, IEE Colloquium on, S. 9/1, 1994.

- [14] Kester, W.: Understand SINAD, ENOB, SNR, THD, THD+ N, and SFDR so you don't get lost in the noise floor. Tutorial, 2009. <http://www.analog.com/media/en/training-seminars/tutorials/MT-003.pdf>, abgerufen am 19.12.2017.
- [15] Klöck, J. F.: Modellierung und Regelung von Transversalflussmaschinen unter Berücksichtigung harmonischer Stromkomponenten. Doktorarbeit, TU Braunschweig, 2016.
- [16] Koch, J. und M. Stämpfle: Mathematik für das Ingenieurstudium. Carl Hanser Verlag GmbH & Co. KG, München, 4 Aufl., 2018.
- [17] Kouvaras, N.: Operations on delta-modulated signals and their application in the realization of digital filters. Radio and Electronic Engineer, 48(9):431–438, 1978.
- [18] Kouvaras, N.: Some novel elements for delta-modulated signal processing. Radio and Electronic Engineer, 51(5):241–250, 1981.
- [19] Kouvaras, N.: Novel multi-input signal-processing networks with reduced quantization noise. International Journal of Electronics, 56(3):371–378, 1984.
- [20] Lagoyannis, D. und E. C. Georgoudis: Multiplier of delta-sigma sequences of the same clock frequency. International Journal of Electronics, 57(3):423–427, 1984.
- [21] Lagoyannis, D. und K. Pekmestzi: Multipliers of delta-sigma sequences. Radio and Electronic Engineer, 51(6):281–286, 1981.
- [22] Leonhard, W.: Regelung elektrischer Antriebe. Springer, Berlin, 2., völlig überarb. und erw. Aufl., 2000.
- [23] Liu, Y., P. M. Furth und W. Tang: Hardware-Efficient Delta Sigma-Based Digital Signal Processing Circuits for the Internet-of-Things. Journal of Low Power Electronics and Applications, 5(4):234, 2015.
- [24] Luckjiff, G. A. und I. Dobson: Hexagonal $\Sigma\Delta$ modulators in power electronics. Power Electronics, IEEE Transactions on, 20(5):1075–1083, 2005.
- [25] Marques, A., V. Peluso, M. S. Steyaert und W. M. Sansen: Optimal parameters for $\Delta\Sigma$ modulator topologies. Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on, 45(9):1232–1241, 1998.
- [26] Martens, O., J. Klöck und W. Schumacher: Evaluation of Multi-Axis Control Systems. In: International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management (PCIM), 2015.

- [27] O’Leary, P. und F. Maloberti: Bit stream adder for oversampling coded data. Electronics Letters, 26(20):1708–1709, 1990.
- [28] Ortmanns, M. und F. Gerfers: Continuous-time sigma-delta AD conversion: Fundamentals, performance limits and robust implementations, Bd. 21 d. Reihe Springer series in advanced microelectronics. Springer, Berlin, 2006.
- [29] Patel, N.: Bit-streams: Applications in Control. Doktorarbeit, University of Auckland, 2006.
- [30] Patel, N., G. Coghill und S. K. Nguang: Digital realization of analogue computing elements using bit streams. In: System-on-Chip for Real-Time Applications, Proceedings. The 3rd IEEE International Workshop on, S. 76–80, 2003.
- [31] Reiss, J. D.: Understanding Sigma-Delta Modulation: Solved and Unsolved Issues. J. Audio Eng. Soc., 56(1/2):49–64, 2008.
- [32] Rüschemdorf, L.: Mathematische Statistik. Springer-Lehrbuch Masterclass. Springer Spektrum, Berlin, 2014.
- [33] Schreier, R.: Delta Sigma Toolbox. Matlab toolbox, 2000. <http://www.mathworks.com/matlabcentral/fileexchange/19-delta-sigma-toolbox>, abgerufen am 21.06.2017.
- [34] Schreier, R. und G. C. Temes: Understanding delta-sigma data converters. IEEE Press, Piscataway, NJ, 2005.
- [35] Schröder, D.: Leistungselektronische Schaltungen: Funktion, Auslegung und Anwendung. Springer-Lehrbuch. Springer Berlin Heidelberg, Berlin Heidelberg, 3. Aufl., 2012.
- [36] Schröder, D.: Elektrische Antriebe - Regelung von Antriebssystemen. Springer Berlin Heidelberg, Berlin, Heidelberg, 4. Aufl. 2015 Aufl., 2015.
- [37] Teodorescu, R., M. Liserre und P. Rodríguez: Grid converters for photovoltaic and wind power systems. Wiley IEEE, Chichester, West Sussex, 2011.
- [38] Vass, M.: Wasseruhr des Ktesibios. Webseite, 2018. <http://www.techkyb.de/allgemeines/historische-entwicklung/wasseruhr-des-ktesibios/>, abgerufen am 30.07.2018.

Veröffentlichungen im Rahmen dieser Dissertation

- [HKKS16] Homann, M. ; Klein, A. ; Kirchner, R. ; Schumacher, W.: Quasi-kontinuierliche Signalverarbeitung mit Delta Sigma Bitströmen in der Antriebstechnik - Ein Überblick. In: Fortschritte in der Antriebs- und Automatisierungstechnik, 2016, S. 215–227
- [HKS16] Homann, M. ; Klein, A. ; Schumacher, W.: Direct Delta Sigma Signal Processing for Control of Power Electronics. In: International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management (PCIM), 2016, S. 216–223
- [Kle18] Klein, Axel: Logic Operators on Delta-Sigma Bit-Streams. In: Mathematical and Computational Applications 23 (2018), Nr. 1
- [KS18] Klein, A. ; Schumacher, W.: Algebraic operations on Delta-Sigma bit-streams. In: Mathematical and Computational Applications 23 (2018), Nr. 3
- [KTHS17] Klein, A. ; Thielmann, M. ; Homann, M. ; Schumacher, W.: Operation Modes of a High Bandwidth Three Phase Hysteresis Modulator based on Delta Sigma Bitstreams. In: 19th European Conference on Power Electronics and Applications (EPE), 2017, S. 1–10
- [KTS18a] Klein, A. ; Thielmann, M. ; Schumacher, W.: Sensorless current and speed control of a PMSM driven with a $\Delta\Sigma$ -PWM. In: 20th European Conference on Power Electronics and Applications (EPE), 2018, S. 1–10
- [KTS18b] Klein, A. ; Thielmann, M. ; Schumacher, W.: Switching Frequency Control for a $\Delta\Sigma$ -PWM. In: International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management (PCIM), 2018, S. 141–148
- [MTK18] Matthes, C. ; Thielmann, M. ; Klein, A.: Ressourceneffiziente Regelungen im FPGA. In: atp edition 60 (2018), Nr. 03, S. 56–65
- [TKHS17] Thielmann, M. ; Klein, A. ; Homann, M. ; Schumacher, W.: Analysis of instantaneous switching frequency of a hystere-

sis based PWM for control of power electronics. In: PCIM Europe 2017; International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management, VDE VERLAG GMBH, 2017, 390–397

A Anhang

A.1 Parameter der Delta-Sigma Modulatoren

Parameter der $\Delta\Sigma$ -M mit minimalen Parametern der Ordnung 2 bis 4 [25]:

$$a_2 = \begin{bmatrix} \frac{1}{16} & \frac{1}{2} \end{bmatrix} \quad (\text{A.1})$$

$$b_2 = \begin{bmatrix} \frac{1}{16} & 0 & 0 \end{bmatrix} \quad (\text{A.2})$$

$$a_3 = \begin{bmatrix} \frac{1}{64} & \frac{1}{8} & \frac{1}{2} \end{bmatrix} \quad (\text{A.3})$$

$$b_3 = \begin{bmatrix} \frac{1}{64} & 0 & 0 & 0 \end{bmatrix} \quad (\text{A.4})$$

$$a_4 = \begin{bmatrix} \frac{1}{1024} & \frac{1}{64} & \frac{1}{8} & \frac{1}{2} \end{bmatrix} \quad (\text{A.5})$$

$$b_4 = \begin{bmatrix} \frac{1}{1024} & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{A.6})$$

Parameter der $\Delta\Sigma$ -M mit Auslegung nach der Toolbox der Ordnung 2 bis 4 (auf 4 Stellen gerundet) [33].

$$a_2 = [0,2164 \quad 0,7749] \quad (\text{A.7})$$

$$b_2 = [0,2164 \quad 0 \quad 0] \quad (\text{A.8})$$

$$a_3 = [0,04398 \quad 0,2881 \quad 0,7997] \quad (\text{A.9})$$

$$b_3 = [0,04398 \quad 0 \quad 0 \quad 0] \quad (\text{A.10})$$

$$a_4 = [0,006142 \quad 0,06471 \quad 0,3085 \quad 0,8055] \quad (\text{A.11})$$

$$b_4 = [0,006142 \quad 0 \quad 0 \quad 0 \quad 0] \quad (\text{A.12})$$

Parameter der $\Delta\Sigma$ -M mit Auslegung nach der Toolbox mit SÜF=1 der Ordnung 2 bis 4 (auf 4 Stellen gerundet) [33].

$$a_2 = [0,2164 \quad 0,7749] \quad (\text{A.13})$$

$$b_2 = [0,2164 \quad 0,7749 \quad 1] \quad (\text{A.14})$$

$$a_3 = [0,04398 \quad 0,2881 \quad 0,7997] \quad (\text{A.15})$$

$$b_3 = [0,04398 \quad 0,2881 \quad 0,7997 \quad 1] \quad (\text{A.16})$$

$$a_4 = [0,006142 \quad 0,06471 \quad 0,3085 \quad 0,8055] \quad (\text{A.17})$$

$$b_4 = [0,006142 \quad 0,06471 \quad 0,3085 \quad 0,8055 \quad 1] \quad (\text{A.18})$$

A.2 Lösung der PWM-Schaltfrequenz

$$\begin{aligned}
 f_{\text{PWM}} &= \left(\sum_{i=1}^6 T_i \right)^{-1} \cdot f_s \\
 &= \frac{2 \cdot v_1^4 \cdot v_2 - v_1^4 \cdot v_{\text{DC}} - 4 \cdot v_1^3 \cdot v_2^2 - 2 \cdot v_1^3 \cdot v_2 \cdot v_{\text{DC}} + 2 \cdot v_1^3 \cdot v_{\text{DC}}^2 + 4 \cdot v_1^2 \cdot v_2^3 + 3 \cdot v_1^2 \cdot v_2^2 \cdot v_{\text{DC}} - v_1^2 \cdot v_{\text{DC}}^3 - 2 \cdot v_1 \cdot v_2^4 - 2 \cdot v_1 \cdot v_2^3 \cdot v_{\text{DC}} + v_2^4 \cdot v_{\text{DC}}}{(3 \cdot v_1^2 \cdot v_2 - 2 \cdot v_1^2 \cdot v_{\text{DC}} - 3 \cdot v_1 \cdot v_2^2 - 2 \cdot v_1 \cdot v_2 \cdot v_{\text{DC}} + 2 \cdot v_1 \cdot v_{\text{DC}}^2 + 2 \cdot v_2^2 \cdot v_{\text{DC}}) \cdot v_{\text{DC}} \cdot N_1} \cdot f_s
 \end{aligned}$$

(A.19)